

THE MODELING AND ANALYSIS
OF AN ELASTIC MECHANISM
WITH CLEARANCES

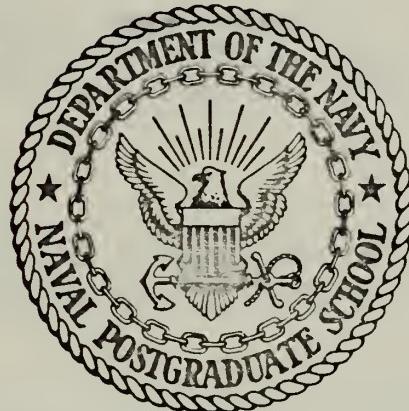
CHARLES WILLIAM GNILKA

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

THE MODELING AND ANALYSIS OF AN ELASTIC
MECHANISM WITH CLEARANCES

by

Charles William Gnilka

Thesis Advisor:

D. Salinas

December 1971

Approved for public release; distribution unlimited.

The Modeling and Analysis of An Elastic
Mechanism With Clearances

by

Charles William Gnilka
Lieutenant, United States Navy
B.S., Villanova University, 1964

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1971

ABSTRACT

The classical approach for the analysis of linkages has been to assume rigid elements during operation. As the operational speeds of mechanisms increase, this assumption creates larger errors in the analysis. Thus a better model of a machine that considers elastic deformations will aid in more efficient design as well as provide improved accuracy and performance.

This thesis presents a simulation for the dynamic response of an elastic link model with three clearances. The simulation has been developed for an automobile, cam actuated valve train. The computer program is coded in FORTRAN IV, for an IBM-360 computer and is included as a portion of this work. The capability to visually observe the dynamic action of the model is included by a graphic display routine. This routine is implemented in SDS FORTRAN IV for a SDS-9300 computer interfaced with an ADAGE Graphic Display Terminal, Model 10.

Sample problems for various cam speeds of interest are included utilizing graphs and photos from the graphic display.

TABLE OF CONTENTS

I.	INTRODUCTION	17
II.	PHYSICAL MODEL	21
A.	GENERAL DESCRIPTION	22
1.	Elements of the Model	22
a.	Follower	22
b.	Left Rocker Arm	22
c.	Right Rocker Arm	22
d.	Valve Spring	23
e.	Valve	23
f.	Cam	23
g.	Valve Seat	23
2.	Clearance Parameters	23
B.	ELEMENT COORDINATES	24
C.	SYSTEM COORDINATES	26
D.	DEFINITIONS OF CONFIGURATIONS	27
1.	External to the System	27
a.	Clearance 1	27
b.	Clearance 3	27
2.	Internal to the System	27
a.	Clearance 2	27
b.	Clearance 4	27
E.	SYSTEM CONFIGURATIONS	28
F.	SHIFTING OF CONFIGURATIONS	32

III.	MATHEMATICAL MODEL	34
A.	EQUATIONS OF MOTION	37
B.	MASS AND STIFFNESS MATRICES	37
C.	DAMPING MATRIX	37
1.	Uncoupling the Equations of Motion	38
2.	Relation of System Parameters to Model Parameters	40
3.	Calculation of System Damping	40
D.	CONFIGURATION SHIFTING ALGORITHM	42
E.	METHOD OF CALCULATIONS	47
1.	Solution to the Governing Equations	48
2.	Precalculated Matrices	54
F.	REFINEMENTS IN THE SOLUTION	56
1.	Incremental Time	56
2.	Impact Forces	56
IV.	IBM-360 COMPUTER PROGRAM	59
A.	GENERAL DESCRIPTION	59
B.	GENERALIZATION MATRICES	59
1.	Matrix CONFIG	60
2.	Matrix IDIM	61
3.	Matrix LAYER	61
4.	Matrix IQ	62
5.	Matrix IDP	63
6.	Matrix ICORD	64
7.	Matrix IMPACT	64
C.	DYNAMIC DIMENSIONING	64
D.	PROGRAM STRUCTURE	68

E. SUBROUTINES - - - - -	68
1. Subroutine THESIS - - - - -	70
2. Subroutine BKSTEP - - - - -	71
3. Subroutines MCK, DVVS, and DINVS - - - - -	71
4. Newmark's Method Subroutines- - - - -	71
a. Subroutine DEL - - - - -	71
b. Subroutine NEWMRK - - - - -	72
5. Subroutines for Shifting Configurations - - - - -	72
a. Subroutine MONTOR - - - - -	72
b. Subroutine DTPRIM - - - - -	72
c. Subroutine SHIFT - - - - -	73
6. Subroutine INPUT - - - - -	73
7. Subroutine CONMAT - - - - -	73
8. Subroutine INITAL - - - - -	74
9. Subroutine CAM - - - - -	74
10. Subroutine BRANCH - - - - -	74
11. Subroutine ADJUST - - - - -	75
12. Subroutine FIXFOR - - - - -	75
13. Output Subroutines - - - - -	75
a. Subroutine AGT - - - - -	75
b. Subroutines, ARRAY, GRAPH, GPRINT, OLPLLOT - -	76
c. Subroutines, DRU - - - - -	76
d. Subroutine LIST - - - - -	76
e. Subroutine PRNTMX - - - - -	77

V.	SDS-9300 COMPUTER PROGRAM	95
A.	METHOD OF ELASTICITY CALCULATIONS	95
B.	PROGRAM STRUCTURE	96
1.	General Description	96
2.	Sense Lights	99
a.	Sense Light 1	99
b.	Sense Light 2	99
c.	Sense Light 4	99
d.	Sense Light 6	99
C.	MAIN PROGRAM	100
D.	SUBROUTINES	100
1.	Subroutine INITAL	101
2.	Subroutine READ	101
3.	Subroutine SPRCTR	102
4.	Subroutine MIDBAR	102
5.	Subroutine DISPLAY	102
6.	Subroutine CAM	102
7.	Subroutine BAR1	103
8.	Subroutine BAR2	103
9.	Subroutine BAR3	103
10.	Subroutine BAR4	103
11.	Subroutine JOINT	104
VI.	NUMERICAL RESULTS	105
A.	MODE SHAPES AND NATURAL FREQUENCIES	106
B.	PRE-CALCULATED SYSTEM FREQUENCIES	107

C. SAMPLE PROBLEMS - - - - -	132
1. Example 1A, THREE CLEARANCES, 11,000 RPM, VALVE SEAT - - - - -	133
2. Example 1B, THREE CLEARANCES, 11,000 RPM, NO VALVE SEAT - - - - -	150
3. Example 1C, TWO CLEARANCES, 11,000 RPM, VALVE SEAT - - - - -	164
4. Example 1D, TWO CLEARANCES, 11,000 RPM, NO VALVE SEAT - - - - -	178
5. Example 2 & 3, THREE CLEARANCES, 9,000 & 10,000 RPM, VALVE SEAT - - - - -	187
VII. CONCLUSION - - - - -	188
APPENDIX A MASS AND STIFFNESS MATRICES - - - - -	190
APPENDIX B NEWMARK'S 3 PARAMETER METHOD - - - - -	196
APPENDIX C FOUR CLEARANCE PROBLEM - - - - -	199
APPENDIX D COMPUTER CONTROL CARDS - - - - -	202
APPENDIX E IBM-360 COMPUTER PROGRAM DATA CARDS - - - - -	209
APPENDIX F GRAPHS FOR EXAMPLES 2 and 3 - - - - -	210
IBM COMPUTER PROGRAM OUTPUT - - - - -	229
IBM-360 COMPUTER PROGRAM LISTING - - - - -	230
SDS-9300 COMPUTER PROGRAM LISTING - - - - -	311
LIST OF REFERENCES - - - - -	338
INITIAL DISTRIBUTION LIST - - - - -	340
FORM DD 1473 - - - - -	341

LIST OF TABLES

- II.1 CONFIGURATION REFERENCE TABLE
- II.2 CONFIGURATION REFERENCE TABLE (continued)
- VI.1 GRAPH REFERENCE POINTS AND DEGREE OF CAM ROTATION FOR FIGURES 6.27 - 6.32
- VI.2 GRAPH REFERENCE POINTS AND DEGREES OF CAM ROTATION FOR FIGURES 6.35 - 6.39
- VI.3 GRAPH REFERENCE POINTS AND DEGREES OF CAM ROTATION FOR FIGURES 6.41 - 6.45
- VI.4 GRAPH REFERENCE POINTS AND DEGREES OF CAM ROTATION FOR FIGURES 6.48 - 6.50

LIST OF FIGURES

- 2.1 CAM ACTUATED VALVE TRAIN
- 2.2 AXIAL ELEMENT
- 2.3 BENDING ELEMENT
- 2.4 COMBINED ELEMENT COORDINATES
- 2.5 FULL SYSTEM COORDINATES
- 2.6 CONFIGURATION SHIFTING DIAGRAM
- 3.1 CONFIGURATION MATRIX
- 3.2 SHIFTING EXAMPLE NO. 1
- 3.3 SHIFTING EXAMPLE NO. 2
- 3.4 MATRIX LAYERS
- 3.5 INCREMENTAL TIME Δt
- 3.6 SYSTEM CURVES FOR IMPACT CONSIDERATIONS
- 4.1 MATRIX CONFIG
- 4.2 MATRIX IDIM
- 4.3 MATRIX LAYER
- 4.4 MATRIX IQ
- 4.5 MATRIX IDP
- 4.6 MATRIX ICORD
- 4.7 DYNAMIC DIMENSIONING OF ARRAY SYSM
- 4.8 DYNAMIC DIMENSIONING OF ARRAY BMIV
- 4.9 IBM-360 COMPUTER PROGRAM STRUCTURE
- 4.10 SUBROUTINE THESIS FLOW DIAGRAM
- 4.11 SUBROUTINE BKSTEP FLOW DIAGRAM
- 4.12 SUBROUTINE DEL FLOW DIAGRAM
- 4.13 SUBROUTINE NEWMRK FLOW DIAGRAM
- 4.14 SUBROUTINE MONTOR FLOW DIAGRAM

- 4.15 SUBROUTINE CAM FLOW DIAGRAM
- 4.16 SUBROUTINE DTPRIM FLOW DIAGRAM
- 4.17 SUBROUTINE INPUT FLOW DIAGRAM
- 4.18 SUBROUTINE CONMAT FLOW DIAGRAM
- 4.19 SUBROUTINE BRANCH FLOW DIAGRAM
- 4.20 SUBROUTINE ADJUST FLOW DIAGRAM
- 4.21 SUBROUTINE FIXFOR FLOW DIAGRAM
- 4.22 SUBROUTINE AGT FLOW DIAGRAM
- 4.23 SUBROUTINE ARRAY FLOW DIAGRAM
- 4.24 SUBROUTINE GRAPH FLOW DIAGRAM
- 4.25 SUBROUTINE GPRINT FLOW DIAGRAM
- 5.1 SDS-9300 COMPUTER PROGRAM STRUCTURE
- 5.2 OVER LAY STRUCTURE
- 6.1 FULL SYSTEM MODE SHAPES AND NATURAL FREQUENCIES
- 6.2 REDUCED SYSTEM MODE SHAPES AND NATURAL FREQUENCIES
- 6.3 CONFIGURATION I - Mass, Damping, and Stiffness
- 6.4 CONFIGURATION I - Precalculated Matrices; BMIV, DA, and DV
- 6.5 CONFIGURATION II - Mass, Damping, and Stiffness
- 6.6 CONFIGURATION II - Precalculated Matrices; COLM, COLC, and COLK
- 6.7 CONFIGURATION II - Precalculated Matrices; BMIV, DA, and DV
- 6.8 CONFIGURATION III - Mass, Damping, and Stiffness
- 6.9 CONFIGURATION III - Precalculated Matrices; COLM, COLC, and COLK

- 6.10 CONFIGURATION IV - Mass, Damping, and Stiffness
- 6.11 CONFIGURATION IV - Precalculated Matrices; COLM, COLC, and COLK
- 6.12 CONFIGURATION IV - Precalculated Matrices; BMIV, DA, and DV
- 6.13 CONFIGURATION V - Mass, Damping, and Stiffness
- 6.14 CONFIGURATION V - Precalculated Matrices; COLM, COLC, and COLK
- 6.15 CONFIGURATION V - Precalculated Matrices; BMIV, DA, and DV
- 6.16 CONFIGURATION VI - Mass, Damping, and Stiffness
- 6.17 CONFIGURATION VI - Precalculated Matrices; COLM, COLC, and COLK
- 6.18 CONFIGURATION VI - Precalculated Matrices; COLM, COLC, and COLK
- 6.19 CONFIGURATION VI - Precalculated Matrices; BMIV, DA, and DV
- 6.20 CONFIGURATION VII - Mass, Damping, and Stiffness
- 6.21 CONFIGURATION VII - Precalculated Matrices; COLM, COLC, and COLK
- 6.22 CONFIGURATION VII - Precalculated Matrices; BMIV, DA, and DV
- 6.23 CONFIGURATION VIII - Mass, Damping, and Stiffness
- 6.24 CONFIGURATION VIII - Precalculated Matrices; COLM, COLC, and COLK
- 6.25 CONFIGURATION VIII - Precalculated Matrices; BMIV, DA, and DV
- 6.26 SAMPLE NUMERICAL RESULTS
- 6.27 - 6.34 EXAMPLE 1A, THREE CLEARANCES, 11,000 RPM, VALVE SEAT
- 6.35 - 6.40 EXAMPLE 1B, THREE CLEARANCES, 11,000 RPM, NO VALVE SEAT

- 6.41 - 6.47 EXAMPLE 1C, TWO CLEARANCES, 11,000 RPM,
VALVE SEAT
- 6.48 - 6.51 EXAMPLE 1D, TWO CLEARANCES, 11,000 RPM,
NO VALVE SEAT
- A.1 FULL SYSTEM MASS MATRIX
- A.2 FULL SYSTEM STIFFNESS MATRIX
- A.3 REDUCED SYSTEM MASS MATRIX
- A.4 REDUCED SYSTEM STIFFNESS MATRIX
- B.1 TIME STEP ACCELERATION CURVE
- C.1 FOUR CLEARANCE SYSTEM CONFIGURATIONS
- C.2 FOUR CLEARANCE SYSTEM CONFIGURATION SHIFTING DIAGRAM
- D.1 INITALIZE DISK SPACE
- D.2 LOAD SUBROUTINE ONTO DISK
- D.3 LIST DATA SET VTOC AND PDS
- D.4 EXECUTE PROGRAM WITH SUBROUTINES ON DISK
- D.5 LOAD MAIN PROGRAM ONTO DISK
- D.6 EXECUTE ENTIRE PROGRAM FROM DISK
- D.7 READ DATA SET MEMBERS ONTO SCRATCH DISK
- D.8 COPY ALL DATA SET MEMBERS
- D.9 COPY PART OF DATA SET MEMBERS
- D.10 TRANSFERRING FORTRAN SOURCE CODE FROM CARDS TO MAGNETIC TAPE
- D.11 PROGRAM EXECUTION FROM FORTRAN OBJECT ON MAGNETIC TAPE
- D.12 PLACING PROGRAM AS BINARY CODE ON MAGNETIC TAPE
- D.13 PROGRAM EXECUTION FROM BINARY CODE ON MAGNETIC TAPE
- F.1 - F.8 EXAMPLE 2, THREE CLEARANCES, 9,000 RPM, VALVE SEAT
- F.9 - F.16 EXAMPLE 3, THREE CLEARANCES, 10,000 RPM, VALVE SEAT

LIST OF PHOTOGRAPHS

- 1.1 AGT Display
- 5.1 AGT Listing Question
- 6.1 AGT Title Page
- 6.2 - 6.23 Example 1A, Three Clearances, 11,000 RPM,
Valve Seat
- 6.24 - 6.42 Example 1B, Three Clearances, 11,000 RPM,
No Valve Seat
- 6.43 - 6.56 Example 1C, Two Clearances, 11,000 RPM,
Valve Seat
- 6.57 - 6.68 Example 1D, Two Clearances, 11,000 RPM,
No Valve Seat

LIST OF SYMBOLS

A	Area
[c]	System damping matrix
c_{ij}	Element of damping matrix
E	Elastic modulus
F_i	Forces
{F*}	Normalized force vector
g	Gravitational Constant
I	Moment of inertia
[K]	System stiffness matrix
k_{ij}	Element of stiffness matrix
L	Length
[M]	System mass matrix
M	Mass
m_{ij}	Elements of mass matrix
{p}	Force vector used in integration solution
q	System coordinate
t	Time
ΔT	Time increment used in integration solution
T_i	Period
u	Element coordinate
{ α }	Vector used in integration solution
[β]	Transformation matrix
ζ	Damping coefficient
	Cam angle of rotation

{Φ}	Eigenvectors listed in columns, modal matrix
{Ψ}	Vector used in integration solution
ω_n, Ω	Natural frequency
< >	Row Vector

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to his advisors, Dr. Richard C. Winfrey currently at the Naval Civil Engineering Laboratory, Port Hueneme, who introduced this author to the project and for his constructive supervision; and Professor David Salinas of the Naval Postgraduate School for his moral support and guidance.

The Naval Postgraduate School Computer Center provided facilities for the computer work, except the computer graphic portion. This portion was accomplished with facilities and technical advice provided by the Electrical Engineering Department Computer Laboratory. In this regard, the author wishes to express his sincere thanks to Mr. Robert L. Limes, computer laboratory Supervisor, and the staff Mr. William A. Thomas and Mr. Albert Wong for the hours in assistance with graphic computer programing and hardware difficulties.

Finally, the author wishes to acknowledge the encouragement, patience, and moral support of his wife, Martha and for her assistance in preparing the manuscript.

I. INTRODUCTION

The classical approach to the dynamic analysis of linkages has been to assume that all members of the machine move without deformation during operation, i.e. that they move as rigid bodies. Based on this assumption, material choices and sizing to withstand stresses for the applied loads, the linkage deformations and the response motion have been severely idealized. Application of factors of safety have been necessary to account for this idealization. At lower operational speeds design criteria are met. However as the speeds increase the resulting large inertial forces and elastic deflections are more severe and thus the possibility of failure more acute. The design becomes invalid when the critical speed is reached where the machine either fails or where it no longer performs its intended task. Thus expensive and time consuming models must be built and tested, for elastic deformations over the range of operational speeds before the kinematic analysis can be fully accepted. The goal then is to obtain a more accurate dynamic model of a machine in order to better predict and understand its performance.

With the advent of the digital computer, more complex mathematical models of machine are being developed. Engineers are able to accomplish more refined designs with a higher degree of operational confidence, less of a factor of safety, better material choice, and fewer physical model experiments.

This work presents a simulation for the dynamic response of a planar elastic link mechanism, free to move in its rigid body motion

as well as its elastic motion. The structural elements deform elastically due to forces applied by neighboring elements, and due to inertial forces. This simulation has been developed for an automobile, cam actuated valve train, with three clearances. The clearances are considered to exist at the (1) cam-pushrod, (2) valve-valve seat, and (3) rocker arm-pushrod. To simplify the required derivations, the model is assumed to initially consist of straight members with constant circular cross-sectional bars. The assumption of constant circular cross-sectional bars is selected for ease in computing beam section properties. The use of straight links greatly simplifies the equations of elastic deformation.

An existing computer program written by Anderson and Winfrey (Ref. 2), was dependent on separately coded sections for each physical configurations of the two clearance model of a cam actuated valve train. As the number of system clearances are increased, the number of system configurations increase exponentially and separately coded program sections become infeasible.

The extent of this author's objectives and work on the existing program is to:

- (1) Restructure and reprogram the simulation to provide for a generalized method for establishing the physical model configurations.
- (2) Modify this program to allow for more than two clearances, i.e. add clearances internal to the system.
- (3) Establish a method for the dynamic graphical display of the model showing the resulting elastic deformations of the model members at high operational speeds.

Objectives (1) and (2) have been accomplished using the IBM-360 computer for the simulation. Objective (3) has been accomplished with the aide of the ADAGE Graphics Terminal model 10 (AGT-10) interfaced with a Scientific Data System computer model 9300 (Photo 1.1). Data generated by the IBM computer is loaded into the SDS computer by means of punched cards or magnetic tape. This data is then manipulated into a form acceptable for display by the AGT-10.

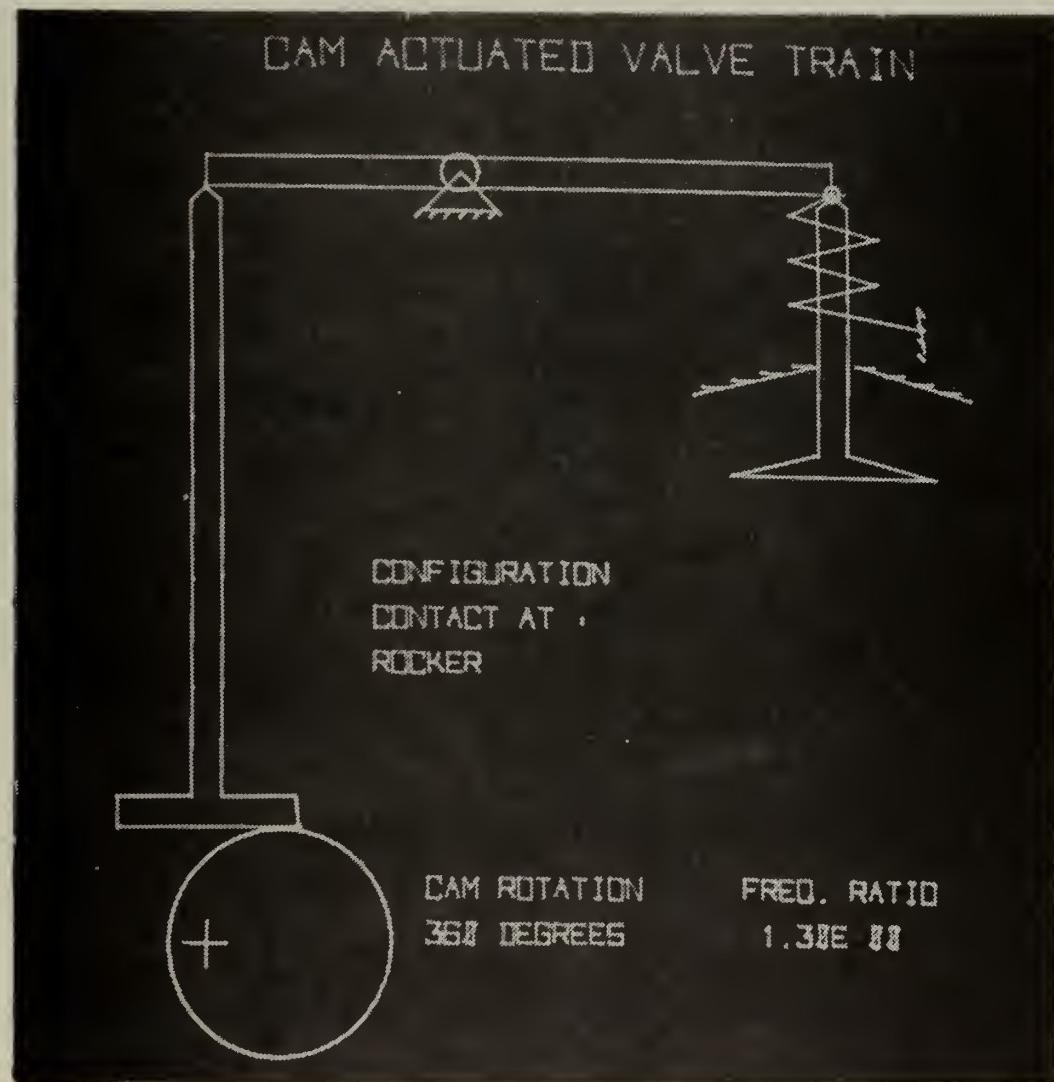


Photo 1.1 AGT DISPLAY

In the following chapters, the physical model, and the concepts and formulation for the corresponding mathematical model are described. The dynamic analysis and the elastic deformation computer programs are discussed and listed at the end of this work. Finally, several numerical examples are given with the model responses shown as AGT photographs and CALCOMP graphs.

In the following chapters, the physical model, and the concepts and formulation for the corresponding mathematical model are described. The dynamic analysis and the elastic deformation computer programs are discussed and listed at the end of this work. Finally, several numerical examples are given with the model responses shown as AGT photographs and CALCOMP graphs.

II. PHYSICAL MODEL

In this chapter the physical system, Figure 2.1, is idealized to enable formulation of a mathematical model whose analysis is tractable. The description covers the interaction between the system parameters and the physical constraints that limit the model's motion. The physical elements that comprise the system are listed and their functions explained. Additionally their physical properties, size and material content, are noted.

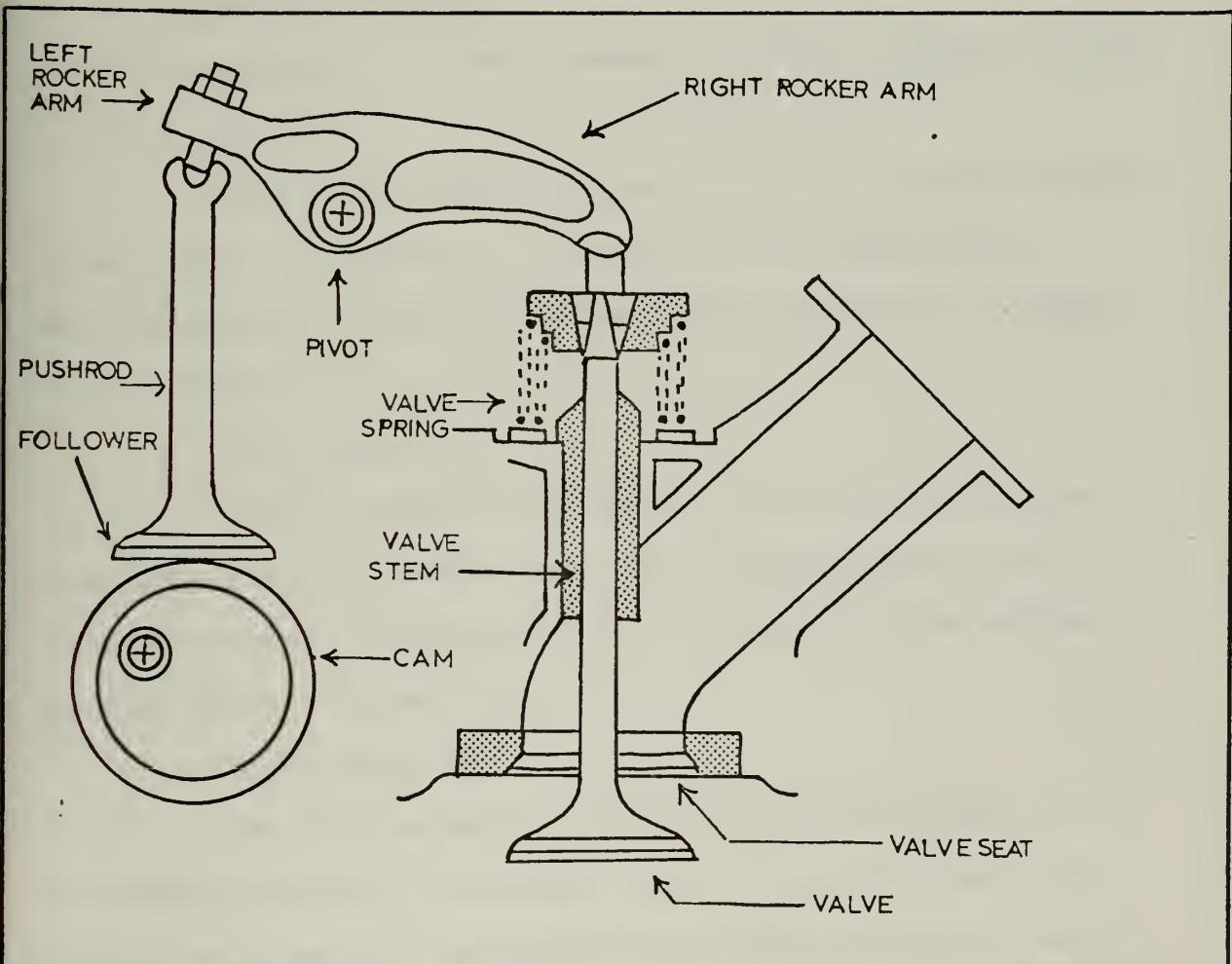


Figure 2.1 CAM ACTUATED VALVE TRAIN

A. GENERAL DESCRIPTION

In order to relate the formulation of this mathematical model to the real world, an automobile car engine valve train assembly is considered. However, any similar physical model might have been chosen. All members of the model are assumed to be manufactured from steel with an elastic modulus of 30×10^6 pounds per square inch and a specific weight of .283 pounds per cubic inch. Additionally longitudinal members are considered as rods with circular cross sections. The diameter and length of each bar are specified.

1. Elements of the Model

The elements of the valve assembly pictured in Figure 2.1 are:

a. Follower

This reciprocating flat faced follower is rigidly attached to the pushrod. The pushrod is considered as an elastic member. These members have harmonic displacement. The diameter is .3 inches; length 9 inches.

b. Left Rocker Arm

The left rocker arm is pinned to and rotates about the fixed pivot point at its right end. This element provides for an "internal" clearance between itself and the pushrod at its left end. The diameter is .6 inches; length 1 inch.

c. Right Rocker Arm

The right rocker arm is also pinned to and rotates about the fixed pivot point. At its right end it is pinned to the valve stem and valve spring. The diameter is .6 inches; length 1.5 inches.

d. Valve Spring

The free end of the valve spring is connected to the rocker arm and valve stem pin, the other end is rigidly fixed to ground. The spring is assumed to have a spring constant of 245 pounds per inch. The diameter is .07 inches; free length 36 inches.

e. Valve

This reciprocating valve is rigidly attached to the valve stem. The valve stem is assumed to be elastic. The upper end of the valve stem is connected to the common pin with the valve spring and right rocker arm. The diameter is .3 inches; length 3 inches.

To complete the model two additional elements are added external to the system. Both provide for the "external" clearances. They are:

f. Cam

The offset circular disk cam imparts sinusoidal motion to the follower.

g. Valve Seat

This rigid seating surface for contact by the valve may be removed from the system when desired.

The rocker arm, although modeled as two elements, is considered to be one continuous member. Thus only rotation about the fixed pivot point is permitted.

2. Clearance Parameters

The clearance or interference at the valve seat may be eliminated by the introduction of a predetermined parameter. Due to the choice of the coordinate system, the parameter set less than the minimum value of the valve displacement has the effect of removing

the valve seat from the system. This provides for the flexibility to conduct studies of the valve motion unimpeded by the valve seat.

Additional flexibility of the model is provided by introduction of another clearance parameter. This parameter has the effect of eliminating the internal clearance between the pushrod and left rocker arm. Thus these two members can be pinned together when desired.

The circular disk cam rotates counterwise. This motion first lifts the follower by direct contact to its extreme position and next permits the follower to return to its initial position. The harmonic cycle is then repeated. For systems with cams of this type, the return motion of the follower must be accomplished by some force external to the cam. Therefore the valve spring is introduced to supply this force. For this work, simple harmonic motion is specified for the follower. The follower then oscillates about the midstroke position with an acceleration proportional to its distance from this position. The amplitude is half the stroke. The cam profile used to obtain the harmonic motion has been generated by the equation $.5 \sin(\omega t)$ inches, where ω is the cam speed in radians per second. Similar studies of various other cam profiles are discussed in Refs.

14-18.

B. ELEMENT COORDINATES

The model of the mechanism consists of two basic elements which account for all the relevant energy. The axial element, Figure 2.2 considers tension and compression forces. The bending element, Figure 2.3, considers only bending energy. The coordinates of the five

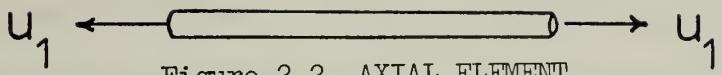


Figure 2.2 AXIAL ELEMENT

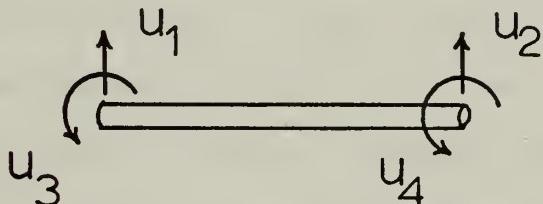


Figure 2.3 BENDING ELEMENT

elements of the system, i.e. pushrod, rocker arms, valve, and spring are chosen as indicated in Figure 2.4.

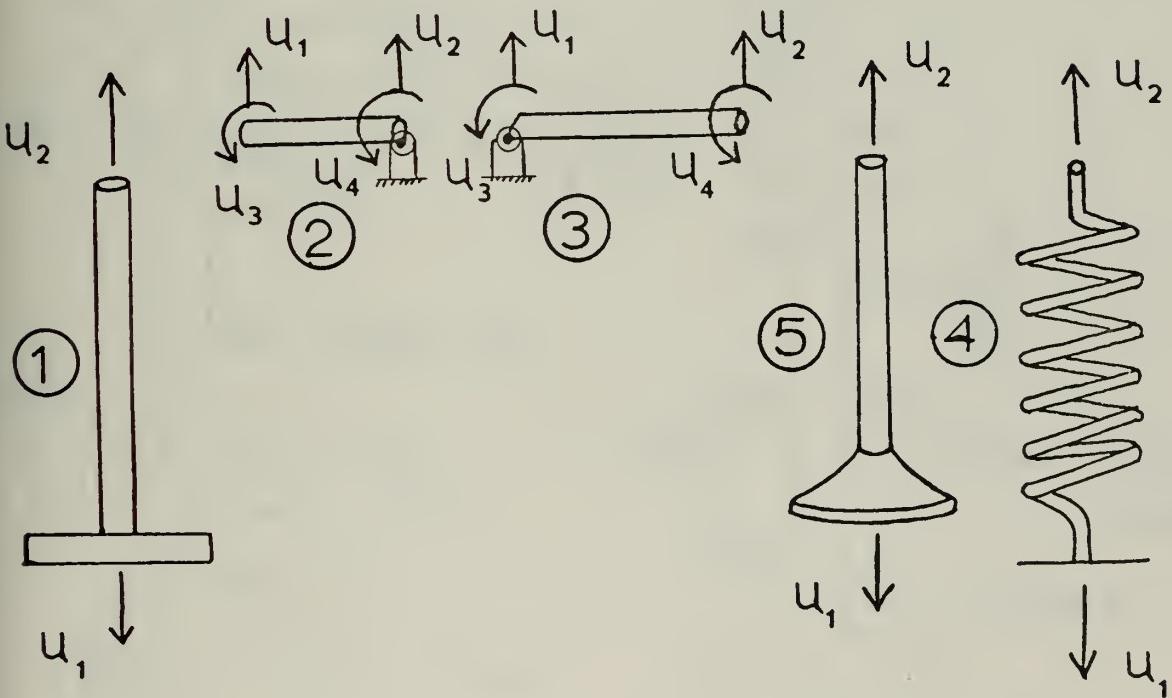


Figure 2.4 COMBINED ELEMENT COORDINATES

C. SYSTEM COORDINATES

Assuming small deformation theory, the linear model is composed of the above elements by superposition. The external members, the cam and valve seat, are also assigned coordinates for reference. As noted above, the pivot point and lower end of the spring are connected to ground and are therefore rigid. The systems coordinates $\{q\}$ define motion. Since no motion is assumed for these two connections, no coordinates are necessary. Recalling that each member was assumed straight, the system coordinates will generally coincide with the element coordinates. The coordinate indices and their positive senses are shown in Figure 2.5.

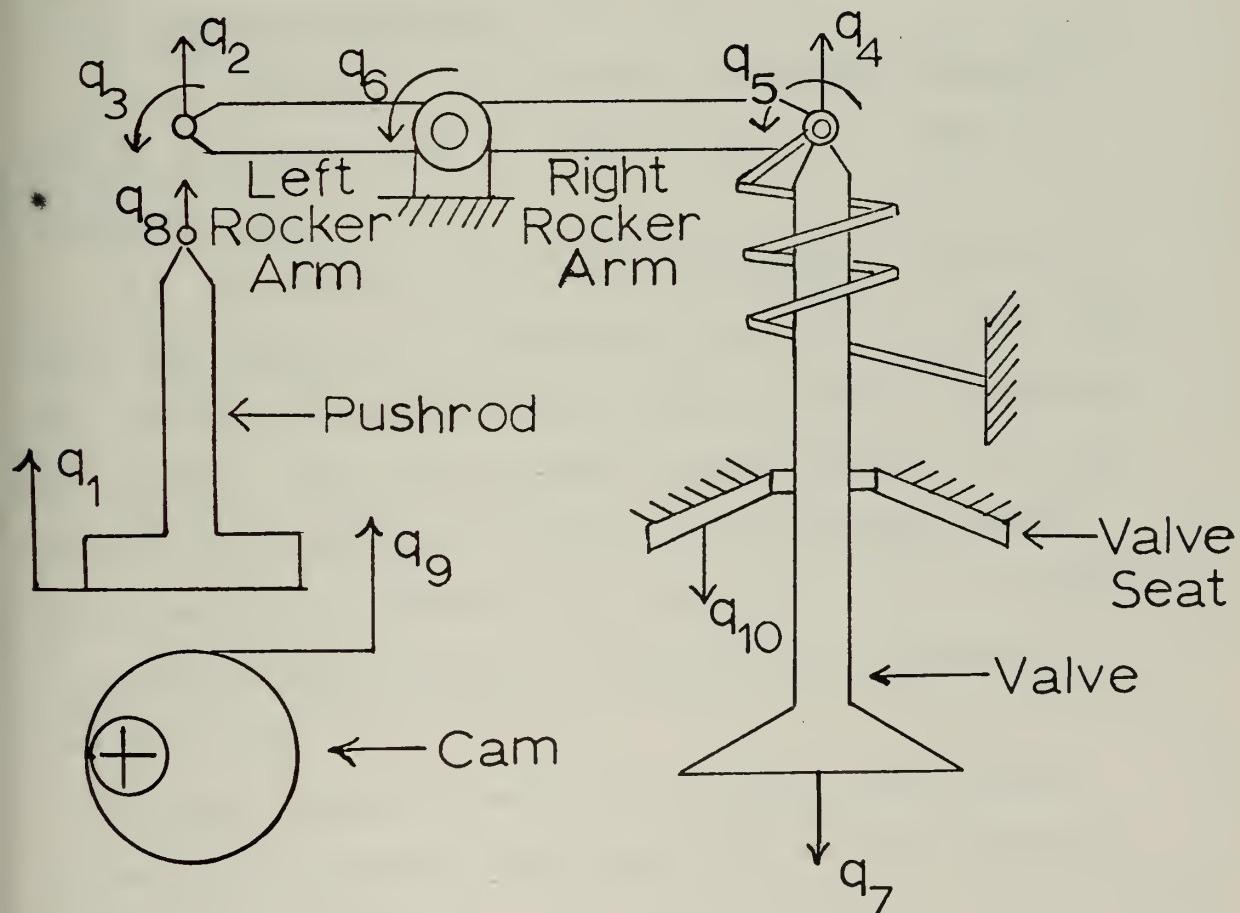


Figure 2.5 FULL SYSTEM COORDINATES

D. DEFINITIONS OF CONFIGURATIONS

The model shown in Figure 2.5 has two external clearances and one internal clearance. The term external refers to a displacement difference between a member of the system and an external member. Internal clearances are displacement differences between two internal members of the system. The clearances are:

1. External To The System

a. Clearance 1. The displacement difference between the cam, (coordinate nine), and the follower, (coordinate one).

b. Clearance 3. The displacement difference between the valve, (coordinate seven), and the valve seat, (coordinate ten).

2. Internal To The System

a. Clearance 2. The displacement difference between the pushrod, (coordinate eight) and left end of the left rocker arm, (coordinate two).

b. Clearance 4. The displacement difference between the valve stem, free end of the spring, and right end of the right rocker arm. The basic concept for the four clearance problem is presented in Appendix C. Since only the three clearance problem is considered for this work, clearance four is zero. Thus the members remain pinned and all move together at this connection. Only one coordinate is required to model the motion of the joint and is designated coordinate four.

Thus the external clearances define positions where external forces must be considered, i.e. forces associated with coordinate

one and coordinate seven. The internal clearances produce internal forces. These equal and opposite forces act upon the coordinate pair two and eight only when they are in contact.

The three clearances, where positive displacement differences between the coordinate pairs may exist, will be referred to as noted above, i.e. clearances 1, 2, and 3. Since clearance 4 will not be modeled, reference to this position will be referred to as coordinate four. Forces between clearances will be discussed in section 3.E.

E. SYSTEM CONFIGURATION

Depending upon the value of the force and displacement at these three clearances, the model can exist in one of eight possible configurations. If clearance 2 is considered pinned, the difference between coordinates two and eight is zero. The possible configurations is then reduced to four and is the study conducted by Anderson (Ref. 2).

The eight configurations are diagramed in Table 2.1 and 2.2.
They are:

1. No contact at clearance 1, clearance 2, or clearance 3.
2. Contact at clearance 3; no contact at clearance 1, or clearance 2.
3. Contact at clearance 2; no contact at clearance 1, or clearance 3.
4. Contact at clearance 2, and clearance 3; no contact at clearance 1.
5. Contact at clearance 1; no contact at clearance 2, and clearance 3.

TABLE 2.1 CONFIGURATION REFERENCE TABLE

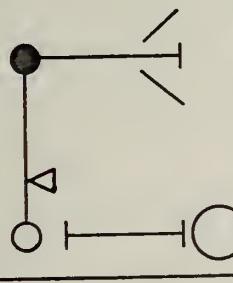
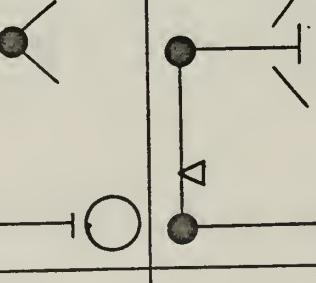
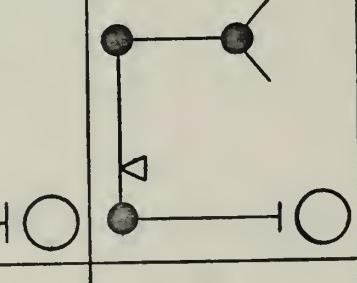
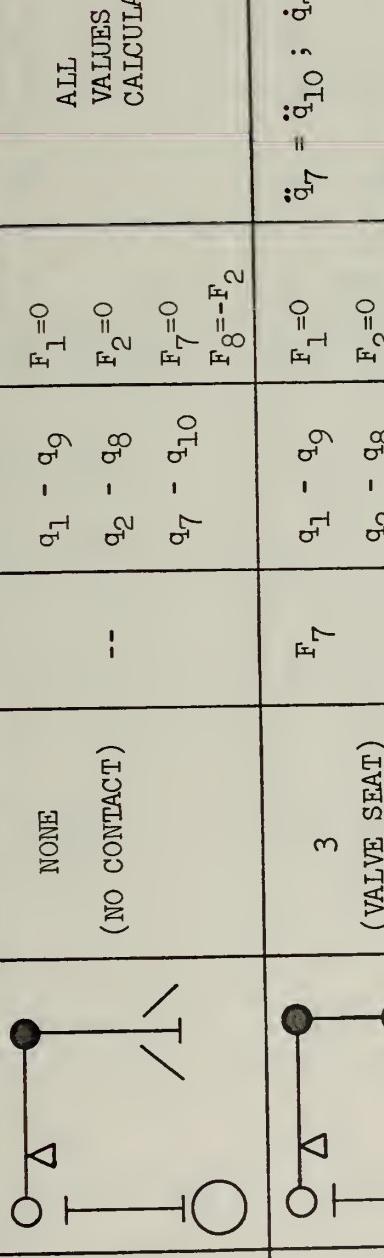
CONFIGURATION		CONTACT AT CLEARANCE		MONITORED VALUES		ASSUMED VALUES	
No.	Figure			Force	DISPL DIFF	Force	Displacements
I		NONE (NO CONTACT)	--	$q_1 - q_9$ $q_2 - q_8$ $q_7 - q_{10}$	$F_1 = 0$ $F_2 = 0$ $F_7 = 0$ $F_8 = -F_2$	$\ddot{q}_7 = \ddot{q}_{10}$; $\dot{q}_7 = \dot{q}_{10}$; $q_7 = q_{10}$	ALL VALUES CALCULATED
II		3 (VALVE SEAT)	F_7	$q_1 - q_9$ $q_2 - q_8$	$F_1 = 0$ $F_2 = 0$ $F_8 = -F_2$	$\ddot{q}_2 = \ddot{q}_8$; $\dot{q}_2 = \dot{q}_8$; $q_2 = q_8$.	∴ Seven Coordinate Reduced System
III		2 (ROCKER)	F_2	$q_1 - q_9$	$F_1 = 0$	$\ddot{q}_2 = \ddot{q}_8$; $\dot{q}_2 = \dot{q}_8$; $q_2 = q_8$.	∴ Seven Coordinate Reduced System
IV		2 and 3 (VALVE SEAT ROCKER)	F_2 F_7	$q_1 - q_9$	$F_1 = 0$ $F_8 = -F_2$	$\ddot{q}_2 = \ddot{q}_8$; $\dot{q}_2 = \dot{q}_8$; $q_2 = q_8$.	∴ Seven Coordinate Reduced System

TABLE 2.2 CONFIGURATION REFERENCE TABLE (continued)

CONFIGURATION		CONTACT AT CLEARANCE		MONITORED VALUES		ASSUMED VALUES	
No.	Figure	Force	DISPL DIFF	Force	Displacements		
V		F_1	$q_2 - q_8$ $q_7 - q_{10}$	$F_2 = 0$ $F_7 = 0$ $F_8 = -F_2$	$\ddot{q}_1 = \ddot{q}_9; \dot{q}_1 = \dot{q}_9; q_1 = q_9$		
VI		1 and 3 (CAM VALVE SEAT)	F_1 F_7	$q_2 - q_8$	$F_2 = 0$ $F_8 = -F_2$	$\ddot{q}_1 = \ddot{q}_9; \dot{q}_1 = \dot{q}_9; q_1 = q_9$ $\ddot{q}_7 = \ddot{q}_{10}; \dot{q}_7 = \dot{q}_{10}; q_7 = q_{10}$	Seven Coordinate Reduced System
VII		1 and 2 (CAM ROCKER)	F_1 F_2	$q_7 - q_{10}$	$F_7 = 0$ $F_8 = -F_2$	$\ddot{q}_2 = \ddot{q}_8; \dot{q}_2 = \dot{q}_8; q_2 = q_8$ $\ddot{q}_1 = \ddot{q}_9; \dot{q}_2 = \dot{q}_9; q_1 = q_9$	Seven Coordinate Reduced System
VIII		1, 2, and 3 (CAM ROCKER VALVE SEAT)	F_1 F_2 F_7		$F_8 = -F_2$ - - -	$\ddot{q}_2 = \ddot{q}_8; \dot{q}_2 = \dot{q}_8; q_2 = q_8$ $\ddot{q}_1 = \ddot{q}_9; \dot{q}_1 = \dot{q}_9; q_1 = q_9$ $\ddot{q}_7 = \ddot{q}_{10}; \dot{q}_7 = \dot{q}_{10}; q_7 = q_{10}$	Seven Coordinate Reduced System

6. Contact at clearance 1, and clearance 3; no contact at clearance 2.
7. Contact at clearance 1, and clearance 2; no contact at clearance 3.
8. Contact at clearance 1, clearance 2, and clearance 3.

These eight configurations will be referred to as configurations I through VIII respectively.

The physical constraints of the system require that at any given instant of time, one and only one configuration will be satisfied. The forces and displacement differences at clearances 1, 2, and 3 are monitored to determine which configuration is valid for the calculated values. For example when the model is in a configuration that requires contact at a clearance and the calculated force at this clearance becomes zero (or negative), then contact is lost at that coordinate pair. Similarly, when the model is in a configuration that requires no contact at a clearance and the calculated displacement becomes zero (or negative), an interference exists and contact is regained at the coordinate.

The configuration and the values monitored for the appropriate coordinates are listed in Tables 2.1 and 2.2. Since this model always has contact at coordinate four contact at clearance 2 will be referred to as contact at the "rocker", (between left rocker arm and pushrod).

F. SHIFTING OF CONFIGURATIONS

As discussed above, the model shifts from one configuration to another, depending on first, the configuration the model is presently in; and second, the values of the monitored parameters. This shifting of configurations can be more easily understood by considering the flow diagram in Figure 2.6. The configuration schematic of the two clearance problem of Anderson (Ref. 2) appeared as a square, (face 3-4-8-7). As seen here the schematic of the three clearance problem becomes a cube. The solid lines and adjacent force parameter indicate a shift in the direction of the arrow caused by that particular force parameter. Similarly the dashed lines and adjacent coordinate parameters indicate a shift in the direction of the arrow caused by that particular negative displacement difference. Also indicated in the configuration flow diagram is an impact indicator. When a positive clearance becomes zero (or negative) a rapid change in the associated coordinate acceleration occurs which lends to high forces, i.e. impact. This is further discussed in section 3.F2. Impact is indicated by the rectangular boxes associated with the dashed shift lines. The figures within these boxes are the clearances at which the impact occurs.

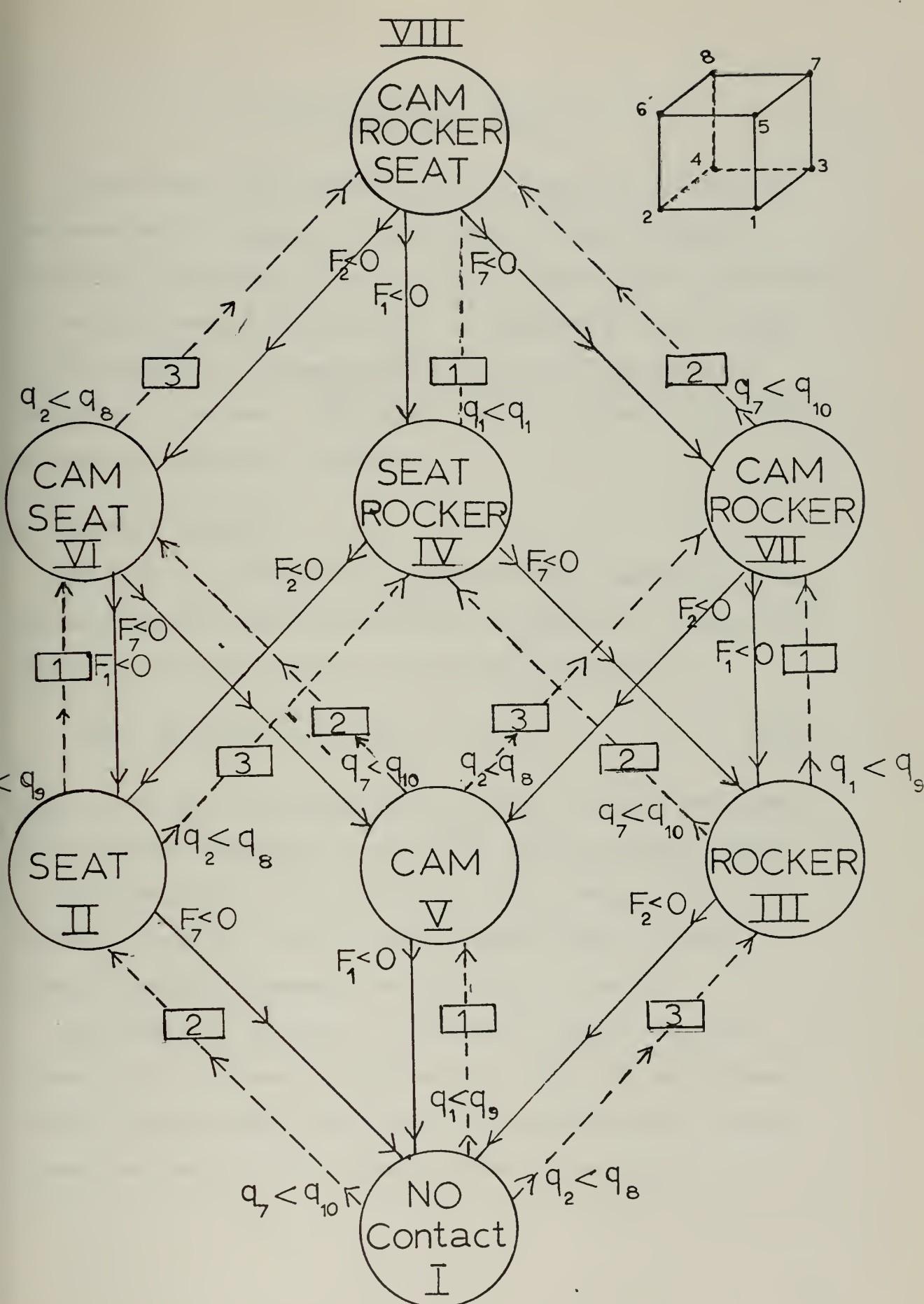


Figure 2.6 FOUR CLEARANCE SYSTEM CONFIGURATION SHIFTING DIAGRAM

III. MATHEMATICAL MODEL

In this chapter the equations of motion which apply to the model are presented and analyzed. The development of the system mass, and stiffness matrices; the two eigenvalue problems; and the formulation of the system damping are discussed. An algorithm to shift the model to the configuration corresponding to the values of the monitored parameters is presented. Finally the numerical methods by which the solutions are obtained is explained.

A. EQUATIONS OF MOTION

The model is considered to be a linear viscous damping spring-mass system, excited by time varying forces. The differential equations of motion can be written in terms of the system coordinates as

$$[M] \{ \ddot{q} \} + [C] \{ \dot{q} \} + [K] \{ q \} = \{ F \} \quad (3.1)$$

where in the usual terminology, $[M]$ represents the system mass matrix, $[C]$ the system damping matrix, and $[K]$ the system stiffness matrix.

The formulation of these matrices has been developed using Lagrange's equations, with the matrix elements based on finite element techniques, Rubenstein and Hurty (Refs. 8 and 9).

The elements of the model, as developed in section 2.B, have a discrete set of element coordinates ($u_i = 1, 2, \dots, n$) that represent the "n", rotational and translational element displacements. These element coordinates $\{u\}$ are directly related to the system

coordinates $\{q\}$ by

$$\{u\} = [\beta] \{q\} \quad (3.2)$$

The transformation matrix $[\beta]$ is formed by applying the conditions of compatibility to the mechanism. For example, if element coordinate u_4 of element three is rotated by one unit, recalling that linear theory is assumed, it is seen that the relation is

$$(u_4)_3 = q_5 \quad . \quad (3.3)$$

This scalar equation viewed as the matrix equation, (3.2), becomes

$$(u_4)_3 = \langle 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \rangle \begin{Bmatrix} q_1 \\ q_2 \\ \vdots \\ q_5 \\ \vdots \\ q_8 \end{Bmatrix}$$

Thus the fifth row of the transformation matrix $[\beta]$ has been obtained. By similar procedures, the entire transformation matrix for the eight coordinate system is developed. Therefore equation (3.2) can now be written as equation (3.4). Where the column matrix of the element coordinates, $\{u\}$, are partitioned by their appropriate elements, and the 14×8 transformation matrix, $[\beta]$ is partitioned into sub-matrices corresponding to the respective elements. A more concise form may be written for equation (3.4) as seen in equation (3.5).

$$\left\{ \begin{array}{l} \{u_1\} \\ \{u_2\} \end{array} \right\}_1 = \left[\begin{array}{ccccccc} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array} \right] = \left\{ \begin{array}{l} \{q_1\} \\ \{q_2\} \\ \{q_3\} \\ \{q_4\} \\ \{q_5\} \\ \{q_6\} \\ \{q_7\} \\ \{q_8\} \end{array} \right\}$$

(3.4)

$$\left\{ \begin{array}{l} \{\mathbf{u}\}_1 \\ \{\mathbf{u}\}_2 \\ \{\mathbf{u}\}_3 \\ \{\mathbf{u}\}_4 \\ \{\mathbf{u}\}_5 \end{array} \right\} = \left[\begin{array}{c} [\beta]_1 \\ [\beta]_2 \\ [\beta]_3 \\ [\beta]_4 \\ [\beta]_5 \end{array} \right] \quad \{\mathbf{q}\}$$

(3.5)

B. MASS AND STIFFNESS MATRICES

The element mass [M_e] and stiffness [K_e] matrices are inherent characteristics of each element and can readily be formulated with finite element methods, (Refs. 1 and 8). The system mass [M] and stiffness [K] matrices can be formed with the aid of the transform matrix [β] as:

$$[M] = \sum_{i=1}^5 [\beta]_i^T [M_e]_i [\beta]_i \quad (3.6)$$

$$[K] = \sum_{i=1}^5 [\beta]_i^T [K_e]_i [\beta]_i \quad (3.7)$$

where $[\beta]_i$ is the i^{th} element submatrix of $[\beta]$; $[K_e]_i$ is the i^{th} element stiffness matrix; and $[M_e]_i$ is the i^{th} element mass. The complete formulation is discussed in appendix A.

C. DAMPING MATRIX

The damping forces and hence system damping matrix is not as easily formulated since these values may depend upon the vibration of the system as well as elements exterior to it. To reduce the complexity of the solution of the differential equations of motion linear viscous damping is assumed.

The system damping is formulated by Hurty and Rubinstein, (Ref. 9) by: (1) solving the eigenvalue problem associated with the equations of motion of the system in undamped free vibrations,

for the modal matrix; (2) using this modal matrix to form the diagonalized system mass and stiffness matrices; (3) from these matrices forming the diagonalized system damping matrix; and (4) using this diagonalized damping matrix to solve for the system damping matrix.

The dynamic model consists of two eigenvalue problems. One eigenvalue problem is for the system of Figure 2.5. Here eight coordinates are required to solve the dynamic equations, since the pushrod and rocker arm are separated, coordinate two and coordinate eight have independent motion. The other eigenvalue problem is associated with the model in contact at clearance 2 and takes the form derived by Anderson (2). When the motion at coordinates two and eight are equal, the eight system equations must be reduced to a lower order system by removal of the redundant equation to facilitate a solution. Thus the system damping matrix, as well as the system mass and stiffness matrices, become order seven. This will be considered in section 3.E.

1. Uncoupling the Equations of Motion

When the differential equations of motion are derived from a set of convenient coordinates, the equations are normally coupled. If the system displacements $\{q\}$ are expressed in terms of the normal modes, the governing equations will uncouple. Thus from the equations of free undamped vibration

$$[M] \{\ddot{q}\} + [K] \{q\} = 0 \quad (3.8)$$

the eigenvalue problem may be formulated as

$$([\omega^2][M] - [K]) \{q\} = \{0\} \quad . \quad (3.9)$$

The solution to this problem will yield the spectral matrix $[\Omega]$

and the modal matrix, $[\Phi]$.

$$[\Omega] = \begin{bmatrix} \omega_1^2 & & 0 \\ & \ddots & \\ & & \omega_n^2 \end{bmatrix} \quad (3.10)$$

$$[\Phi] = [\{\Phi\}_1, \{\Phi\}_2, \dots \{\Phi\}_n] \quad (3.11)$$

The integer "n" is the number of degrees of freedom of the system

i.e. $n = 8$ for the first eigenvalue problem discussed above and

$n = 7$ for the second. The ω_i ($i = 1, 2, \dots, n$) are the natural

frequencies (or eigenvalues) of the system; and the $\{\Phi\}_i$ ($i = 1, 2, \dots, n$)

are the column vectors of mode shapes (or eigenvalues) corresponding

to the natural frequencies. Now the governing coupled equation of

motion for undamped forced vibration,

$$[M] \{\ddot{q}\} + [K] \{q\} = \{F\} \quad (3.12)$$

may be decoupled by performing the coordinate transformation

$$\{q\} = [\Phi] \{\eta\}$$

The dynamic equation can now be written in terms of the modal

coordinates $\{\eta\}$,

$$[\underline{m}] \{\ddot{\eta}\} + [\underline{k}] \{\eta\} = \{F^*\} \quad (3.13)$$

the uncoupled, diagonal mass $[m]$ and stiffness $[k]$ matrices, and the modal force vector $\{F^*\}$, Hurty and Rubinstein, (Ref. 9).

2. Relation of System Parameter To Modal Parameter

The diagonal matrices, $[m]$, $[c]$ and $[k]$ are each related to their corresponding system matrices by transformation with the modal matrix.

$$[m] = [\Phi]^T [M] [\Phi] \quad (3.14)$$

$$[c] = [\Phi]^T [C] [\Phi] \quad (3.15)$$

$$[k] = [\Phi]^T [K] [\Phi] \quad (3.16)$$

Additionally the modal acceleration $\{\ddot{\eta}\}$ and displacement $\{\eta\}$ are related to the system acceleration $\{\ddot{q}\}$ and displacement $\{q\}$ by the modal matrix.

$$\{\ddot{q}\} = [\Phi] \{\ddot{\eta}\} \quad (3.17)$$

$$\{q\} = [\Phi] \{\eta\} \quad (3.18)$$

3. Calculation of the System Damping

The system damping matrix may be obtained from equation (3.15) by matrix multiplication. Thus the diagonal elements of $[c]$ must be calculated before the desired result, $[C]$, can be found. The governing equations are now uncoupled and the uncoupled mass $[m]$ and stiffness $[k]$ matrices are known. Thus the solution formulated for the single degree of freedom system may be applied to the n degree of freedom system.

Equations (3.19) through (3.21), from Thomson (Ref. 3), may be utilized to obtain the desired diagonal elements of the uncoupled damping matrix $[c]$. The natural frequency in radians per second of the undamped oscillation for a single degree freedom system is

$$\omega_n = \sqrt{k/m} \quad (3.19)$$

The damping factor (ζ), is defined as the ratio of the viscous damping (c) to the critical damping factor (c_c).

$$\zeta = c/c_c \quad (3.20)$$

The critical damping factor is also given by

$$c_c = 2 m \omega_n = 2\sqrt{m k} \quad (3.21)$$

Upon substitution of equation (3.21) into equation (3.20), the viscous damping becomes

$$c = 2 \zeta \sqrt{k m} \quad (3.22)$$

In terms of the uncoupled system, the value of the diagonal element of the natural frequencies, corresponding to equation (3.19), are

$$\begin{bmatrix} \omega_1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & \omega_n \end{bmatrix} = \begin{bmatrix} \frac{k_{11}}{m_{11}} & & & \\ & \frac{k_{22}}{m_{22}} & & \\ & & \ddots & \\ & & & \frac{k_{nn}}{m_{nn}} \end{bmatrix} \quad (3.23)$$

where the k 's and m 's are the diagonal elements from $[k]$ and $[m]$.

Similarly equation (3.22) can be written as

$$c_{ii} = 2\zeta_i \sqrt{k_i m_i} \quad (i = 1, 2, \dots n) \quad (3.24)$$

and thus the diagonal values of the uncoupled damping matrix $\begin{bmatrix} c \end{bmatrix}$ are obtained.

An arbitrary choice of values for the damping factor vector $\{\zeta\}$ was considered on the basis of the natural frequencies. The damping factor, (ζ_i), corresponding to the fundamental frequency is set equal to $\frac{1}{2}$. The remaining elements of $\{\zeta\}$ are set to equal $1/10$.

Having established $\begin{bmatrix} c \end{bmatrix}$, the damping matrix $[C]$ may be formed from equation (3.15) by matrix multiplication on the left by $\begin{bmatrix} \Phi^T \end{bmatrix}^{-1}$ and on the right by $\begin{bmatrix} \Phi \end{bmatrix}^{-1}$ as

$$\begin{bmatrix} \Phi^T \end{bmatrix}^{-1} \begin{bmatrix} c \end{bmatrix} \begin{bmatrix} \Phi \end{bmatrix}^{-1} = \begin{bmatrix} \Phi^T \end{bmatrix}^{-1} \begin{bmatrix} \Phi \end{bmatrix}^T [C] \begin{bmatrix} \Phi \end{bmatrix} \begin{bmatrix} \Phi \end{bmatrix}^{-1} \quad (3.25)$$

This gives the desired result

$$[C] = \begin{bmatrix} \Phi^T \end{bmatrix}^{-1} \begin{bmatrix} c \end{bmatrix} \begin{bmatrix} \Phi \end{bmatrix}^{-1} \quad (3.26)$$

D. CONFIGURATION SHIFTING ALGORITHM

For the two clearance problem four configurations were established. This work provided for three clearances having eight possible model configurations. For the addition of the clearance at coordinate four, a total of 4 clearances, there would exist sixteen possible system configurations. Thus the number of possible configurations

of an "n" clearance system would be 2^n . A generalized method of selecting the proper configuration on the basis of the monitored parameters is therefore necessary.

The physical model will shift configurations only on the basis of two general parameters, displacement difference and force. Thus a binary number system facilitates the problem of "computer book-keeping". This is accomplished by assigning the value one to a clearance that is in contact and a zero to a clearance not in contact. For the unique choice of configurations established in chapter 2, Figures 2.4 and 2.5, the configuration matrix is developed as shown in Figure 3.1.

CLEARANCES			
	#1	#2	#3
C	1	0	0
O	0	0	0
N	2	0	0
F	3	0	1
I	3	0	0
G	4	0	1
U	4	0	1
R	5	1	0
A	6	1	0
T	7	1	1
I	7	1	0
O	8	1	1
N			

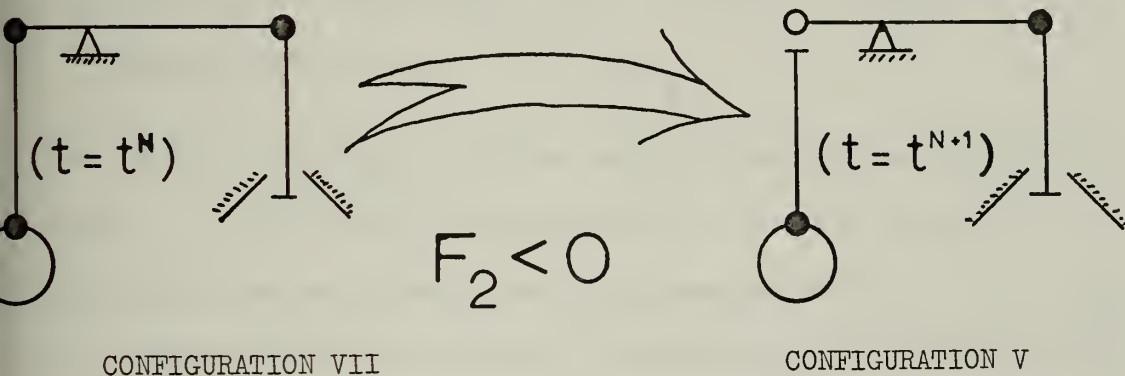
Figure 3.1 Configuration Matrix

The columns of this matrix are associated with clearances 1, 2, and 3 as indicated. The rows are associated with the configuration number that the physical system has taken. Thus to retrieve the correct configuration the entire row is taken as a binary number and converted to a decimal number. Since it is not

convenient to have a configuration numbered zero, a one was arbitrarily added.

For example, when no contact exists at all clearances, the binary number representing this configuration is 0 0 0. This binary number converts to decimal as 0 . Upon adding the above mentioned value one, the decimal value becomes 1, which is the first row of the matrix and hence configuration I.

The binary number 1 1 1 represents contact at all clearances and converts to decimal 7. Adding one yields the value 8, the 8th row of the configuration matrix and hence configuration VIII.



CONFIGURATION VII

CONFIGURATION V

Step 1: VII \rightarrow 1 1 0
 Step 2: $1 \oplus 1 = 1 0$
 Step 3: 1 0 0
 Step 4: 1 0 0 \rightarrow V

\oplus indicates binary addition

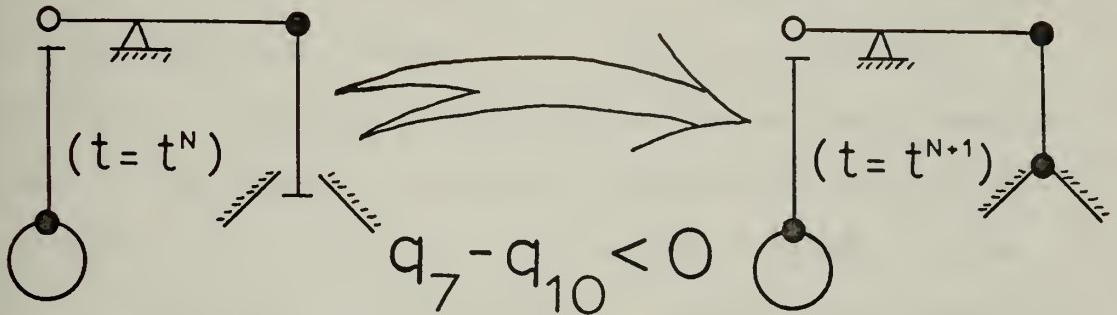
Figure 3.2 Shifting Example No. 1

The manner in which the program determines which configuration the model has shifted into is of primary concern. The developed algorithm is an extension of the above concepts, and best described

by way of examples. For the dynamic behavior pictured in Figure 3.2, the sequence of events is:

- (1) The model is in one of the eight given configurations, i.e. VII at time t^n .
- (2) The program calculates the required parameters for the next time step, t^{n+1} . Among these parameters are the monitored values. In this example the monitored value F_2 has changed during the time step, i.e. F_2 has gone from $F_2 > 0$ to $F_2 < 0$.
- (3) The negative force at coordinate two indicates the loss of contact at clearance 2. Thus the model now shifts to configuration V for time t^{n+1} , (Figure 2.6).

The four step shifting algorithm accomplishes this as outlined in Figure 3.2. The binary representation for configuration VII is retrieved from row seven in the configuration matrix, 1 1 0. Next the column of the binary number corresponding to the clearance that has been altered is removed. In this example, the element being the second digit of the binary number which correspond to the clearance 2 column. A one is then added, by binary addition, to this element. Carries are disregarded. The new second digit 0 is placed into the binary number. Finally the new resulting binary number is converted to decimal and yields the new configuration number, V.



CONFIGURATION V

CONFIGURATION VI

Step 1: $V \rightarrow 1\ 0\ 0$ Step 2: $0 \oplus 1 = 1$ Step 3: $1\ 0\ 1$ Step 4: $1\ 0\ 1 \rightarrow VI$

\oplus indicates binary addition

Figure 3.3 Shifting Example No. 2

The second example for the shifting algorithm is shown in Figure 3.3. The sequence of events for the dynamic behavior is:

- (1) The model is in configuration V at time t^n .
- (2) The program calculates the required parameters at the next time step t^{n+1} . In this example, the monitored displacement difference value $q_7 - q_{10}$ has changed from $q_7 - q_{10} > 0$ to $q_7 - q_{10} < 0$.
- (3) The negative displacement difference at the coordinate pair 7 - 10 indicates that contact is established at clearance 3. Thus the model now shifts to configuration VI for time t^{n+1} .

The shifting algorithm accomplishes this as outlined in Figure 3.3.

The binary representation for configuration V is retrieved from row five in the configuration matrix, 1 0 0. Next the column of the binary number corresponding to the clearance that has been altered is removed. In this example, the element being the third digit of the binary number which corresponds to the clearance 3 column. The new digit, 1 is placed into the binary number. Finally the new resulting binary number is converted to decimal and yields the new configuration number, VI.

E. METHOD OF CALCULATIONS

Newmark's β parameter method, (Ref. 4), forms the basis for the calculations. This one step numerical intergration procedure assumes the accelerations of the system coordinates are a linear function of time. Errors introduced by this assumption are given by $\omega_n \Delta T/6$, where ω_n is the highest natural frequency of interest. For small time steps this error is negligible. In this study the time step is taken as

$$\Delta T = 1 \times 10^{-6} \text{ seconds} \quad (3.27)$$

It will be shown, by example that the solution to the equations of motion takes a convenient form. Precalculated matrix expressions may generally be used, resulting in large savings of computer execution time. A basic familiarity with the formulation of Newmark's method (Appendix B) will prove helpful and aid in the understanding of the following discussion.

The development of the problem calculations will assume the model has been operating in configuration VI, contact at the cam and valve seat. The procedures are identical by extension, to the remaining seven configurations. Thus the problem unknown quantities have been calculated for time $t = t^n$. The object is to solve the governing equations for the next time increment $t = t^{n+1}$.

1. Solution to the Governing Equations

The full system matrices $[M]$, $[C]$, and $[K]$ have been developed in sections 3.B and 3.C. The governing equations of motion are expressed as the matrix equation 3.1 in section 3.A. Written in concise form the governing equations for the eight system coordinates are

$$[\underline{m}_{ij}] \{\ddot{q}_i\}^{n+1} + [\underline{c}_{ij}] \{\dot{q}_i\}^{n+1} + [\underline{k}_{ij}] \{q_i\}^{n+1} = \{F_i\}^{n+1} \\ \cdot (i = 1, 2, \dots, 8; j = 1, 2, \dots, 8) \quad (3.28)$$

Exterior forces, F_1^{n+1} and F_7^{n+1} , are unknown. The remaining applied forces F_i^{n+1} ($i = 2, 3, \dots, 6, 8$) are equal to zero. Since the cam and follower are in contact, the known cam values at reference coordinate nine, \dot{q}_9^{n+1} , \dot{q}_9^{n+1} , and q_9^{n+1} are equal to the corresponding follower values at system coordinate one, \dot{q}_1^{n+1} , \dot{q}_1^{n+1} , and q_1^{n+1} . Similarly, the known values for the stationary valve seat at reference coordinate ten, \dot{q}_{10}^{n+1} , \dot{q}_{10}^{n+1} and q_{10}^{n+1} are set equal to those corresponding values for the valve at system coordinate seven, \dot{q}_7^{n+1} , \dot{q}_7^{n+1} , and q_7^{n+1} . Displacements

q_i^{n+1} ($i = 2, 3, \dots, 6, 8$) are unknown. Expanding and partitioning equations 3.28 to take advantage of the known values yields

$$\begin{bmatrix} m_{11} & m_{12} & \cdots & m_{16} & m_{17} & m_{18} \\ m_{21} & m_{22} & \cdots & m_{26} & m_{27} & m_{28} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ m_{61} & m_{62} & \cdots & m_{66} & m_{67} & m_{68} \\ m_{71} & m_{72} & \cdots & m_{76} & m_{77} & m_{78} \\ m_{81} & m_{82} & \cdots & m_{86} & m_{87} & m_{88} \end{bmatrix} \begin{Bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \vdots \\ \ddot{q}_6 \\ \ddot{q}_7 \\ \ddot{q}_8 \end{Bmatrix} + \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{16} & c_{17} & c_{18} \\ c_{21} & c_{22} & \cdots & c_{26} & c_{27} & c_{28} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ c_{61} & c_{62} & \cdots & c_{66} & c_{67} & c_{68} \\ c_{71} & c_{72} & \cdots & c_{76} & c_{77} & c_{78} \\ c_{81} & c_{82} & \cdots & c_{86} & c_{87} & c_{88} \end{bmatrix} \begin{Bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_6 \\ \dot{q}_7 \\ \dot{q}_8 \end{Bmatrix}$$

$$+ \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{16} & k_{17} & k_{18} \\ k_{21} & k_{22} & \cdots & k_{26} & k_{27} & k_{28} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ k_{61} & k_{62} & \cdots & k_{66} & k_{67} & k_{68} \\ k_{71} & k_{72} & \cdots & k_{76} & k_{77} & k_{78} \\ k_{81} & k_{82} & \cdots & k_{86} & k_{87} & k_{88} \end{bmatrix} \begin{Bmatrix} q_1 \\ q_2 \\ \vdots \\ q_6 \\ q_7 \\ q_8 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ 0 \\ \vdots \\ 0 \\ F_7 \\ 0 \end{Bmatrix} \quad (3.29)$$

The next step creates two sets of equations. The first set, equations 3.30 and 3.31 consists of individual equations in which the displacement, velocity, and accelerations are known but the force is not. The second set, equation 3.32 consists of the matrix equations set equal to the zero force vector. In these equations all displacement, velocity, and acceleration quantities are unknown.

These equations of motion for the system are now written as:

(3.30)

$$F_1 = \langle m_{11} \dots m_{18} \rangle \left(\begin{array}{c} \ddot{q}_1 \\ \vdots \\ \ddot{q}_8 \end{array} \right)^{n+1} + \langle c_{11} \dots c_{18} \rangle \left(\begin{array}{c} \dot{q}_1 \\ \vdots \\ \dot{q}_8 \end{array} \right)^{n+1} + \langle k_{11} \dots k_{18} \rangle \left(\begin{array}{c} q_1 \\ \vdots \\ q_8 \end{array} \right)^{n+1}$$

(3.31)

$$F_7 = \langle m_{71} \dots m_{78} \rangle \left(\begin{array}{c} \ddot{q}_1 \\ \vdots \\ \ddot{q}_8 \end{array} \right)^{n+1} + \langle c_{71} \dots c_{78} \rangle \left(\begin{array}{c} \dot{q}_1 \\ \vdots \\ \dot{q}_8 \end{array} \right)^{n+1} + \langle k_{71} \dots k_{78} \rangle \left(\begin{array}{c} q_1 \\ \vdots \\ q_8 \end{array} \right)^{n+1}$$

(3.32)

$$\boxed{\begin{array}{l} \left[\begin{array}{ccccccccc} m_{22} & m_{23} & \cdots & m_{26} & m_{28} & & & & \\ m_{32} & m_{33} & \cdots & m_{36} & m_{38} & & & & \\ \vdots & \vdots & \ddots & \vdots & \vdots & & & & \\ m_{62} & m_{63} & \cdots & m_{66} & m_{68} & & & & \\ m_{82} & m_{83} & \cdots & m_{86} & m_{88} & & & & \end{array} \right]^{n+1} + \left[\begin{array}{ccccccccc} c_{22} & c_{23} & \cdots & c_{26} & c_{28} & & & & \\ c_{32} & c_{33} & \cdots & c_{36} & c_{38} & & & & \\ \vdots & \vdots & \ddots & \vdots & \vdots & & & & \\ c_{62} & c_{63} & \cdots & c_{66} & c_{68} & & & & \\ c_{82} & c_{83} & \cdots & c_{86} & c_{88} & & & & \end{array} \right]^{n+1} + \left[\begin{array}{ccccccccc} k_{22} & k_{23} & \cdots & k_{26} & k_{28} & & & & \\ k_{32} & k_{33} & \cdots & k_{36} & k_{38} & & & & \\ \vdots & \vdots & \ddots & \vdots & \vdots & & & & \\ k_{62} & k_{63} & \cdots & k_{66} & k_{68} & & & & \\ k_{82} & k_{83} & \cdots & k_{86} & k_{88} & & & & \end{array} \right]^{n+1} \end{array}}$$

$$\left(\begin{array}{c} 0 \\ \vdots \\ 0 \end{array} \right)$$

$$+ \Delta P_1^{n+1} + \Delta P_2^{n+1} =$$

Where

$$\Delta P_1^{n+1} = \begin{Bmatrix} m_{21} \\ m_{31} \\ \vdots \\ m_{61} \\ m_{81} \end{Bmatrix} \ddot{q}_1^{n+1} + \begin{Bmatrix} c_{21} \\ c_{31} \\ \vdots \\ c_{61} \\ c_{81} \end{Bmatrix} \dot{q}_1^{n+1} + \begin{Bmatrix} k_{21} \\ k_{31} \\ \vdots \\ k_{61} \\ k_{81} \end{Bmatrix} q_1^{n+1} \quad (3.33)$$

and

$$\Delta P_2^{n+1} = \begin{Bmatrix} m_{27} \\ m_{37} \\ \vdots \\ m_{67} \\ m_{87} \end{Bmatrix} \ddot{q}_7^{n+1} + \begin{Bmatrix} c_{27} \\ c_{37} \\ \vdots \\ c_{67} \\ c_{87} \end{Bmatrix} \dot{q}_7^{n+1} + \begin{Bmatrix} k_{27} \\ k_{37} \\ \vdots \\ k_{67} \\ k_{87} \end{Bmatrix} q_7^{n+1} \quad (3.34)$$

The set of consistent system equations written in concise form are

$$F_1 = \sum_{i=1}^8 (m_{1i} \ddot{q}_i^{n+1} + c_{1i} \dot{q}_i^{n+1} + k_{1i} q_i^{n+1}) \quad (3.35)$$

$$F_7 = \sum_{i=1}^8 (m_{7i} \ddot{q}_i^{n+1} + c_{7i} \dot{q}_i^{n+1} + k_{7i} q_i^{n+1}) \quad (3.36)$$

and

$$[m_{ij}] \{\ddot{q}_i\}^{n+1} + [c_{ij}] \{\dot{q}_i\}^{n+1} + [k_{ij}] \{q_i\}^{n+1} + \sum_{j=1}^2 \Delta P_j^{n+1} = \{0_i\} \quad (3.37)$$

$$(i = 2, 3, \dots, 6, 8; j = 2, 3, \dots, 6, 8)$$

The next step is the solution of the matrix equation 3.37 by Newmark's method, for the unknown accelerations. By integration the velocities and displacements are written in terms of the accelerations as:

$$\{\dot{q}_i\}^{n+1} = \{\dot{q}_i\}^n + \frac{\Delta T}{2} \{\ddot{q}_i\}^n + \frac{\Delta T}{2} \{\ddot{q}_i\}^{n+1} \quad (3.38)$$

and

$$\{q_i\}^{n+1} = \{q_i\}^n + \Delta T \{\dot{q}_i\}^n + \frac{\Delta T^2}{3} \{\ddot{q}_i\}^n + \frac{\Delta T^2}{6} \{\ddot{q}_i\}^{n+1} \quad (3.39)$$

$$(i = 2, 3, \dots, 6, 8)$$

Let

$$\{\alpha_i\}^n = \{\dot{q}_i\}^n + \frac{\Delta T}{2} \{\ddot{q}_i\}^n \quad (3.40)$$

and

$$\{\beta_i\}^n = \{q_i\}^n + \Delta T \{\dot{q}_i\}^n + \frac{\Delta T^2}{3} \{\ddot{q}_i\}^n \quad (3.41)$$

$$(i = 2, 3, \dots, 6, 8)$$

Substituting for $\{\alpha\}^n$ and $\{\beta\}^n$ equations 3.38 and 3.39 become:

$$\{\dot{q}_i\}^{n+1} = \{\alpha_i\}^n + \frac{\Delta T}{2} \{\ddot{q}_i\}^{n+1} \quad (3.42)$$

$$\{q_i\}^{n+1} = \{\beta_i\}^n + \frac{\Delta T^2}{6} \{\ddot{q}_i\}^{n+1} \quad (3.43)$$

$$(i = 2, 3, \dots, 6, 8)$$

Next substituting for $\{\dot{q}_i\}^{n+1}$ and $\{q_i\}^{n+1}$ into equations 3.37 and factoring yields:

$$\left([m_{ij}] + [c_{ij}] \frac{\Delta T}{2} + [k_{ij}] \frac{\Delta T^2}{6} \right) \{ \ddot{q}_i \}^{n+1} + [c_{ij}] \{ \alpha_i \}^n + [k_{ij}] \{ \beta_i \}^n + \sum_{k=1}^2 \Delta P_k^{n+1} = \{ 0_i \}$$

(i = 2,3 .. 6,8; j = 2,3, .. 6,8)

(3.44)

Let

$$[BMIV_{ij}]^{-1} = [m_{ij}] + [c_{ij}] \frac{\Delta T}{2} + [k_{ij}] \frac{\Delta T^2}{6}$$

(i = 2,3, .. 6,8; j = 2,3, .. 6,8)

(3.45)

Substituting for $[BMIV]^{-1}$ in equation 3.44 and moving the knowns to the right side of the equation yields

$$[BMIV_{ij}]^{-1} \{ \ddot{q}_i \}^{n+1} = -[c_{ij}] \{ \alpha_i \}^n - [k_{ij}] \{ \beta_i \}^n - \sum_{k=1}^2 \Delta P_k^{n+1}$$

(i = 2,3,..6,8 ; j = 2,3,..6,8)

(3.46)

Inverting the matrix $[BMIV]^{-1}$ yields the expression for the unknown accelerations

$$\{ \ddot{q}_i \}^{n+1} = -[BMIV_{ij}] [c_{ij}] \{ \alpha_i \}^n - [BMIV_{ij}] [k_{ij}] \{ \beta_i \}^n - [BMIV_{ij}] \left(\sum_{k=1}^2 \Delta P_k^{n+1} \right)$$

(i = 2,3,..6,8; j = 2,3,..6,8)

(3.47)

In equation 3.47 the non-time function matrix multiplications operations are precalculated as

$$[DA_{ij}] = [BMIV_{ij}] [c_{ij}]$$
(3.48)

and

$$[DV_{ij}] = [BMIV_{ij}] [k_{ij}]$$
(3.49)

(i = 2,3,..6,8; j = 2,3,..6,8)

The system accelerations, except the initial known values, are now calculated by

$$\{\ddot{q}_i\}^{n+1} = -[\text{DA}_{ij}] - [\text{DV}_{ij}] - [\text{BMIV}_{ij}] \left(\sum_{k=1}^2 \Delta P_k^{n+1} \right) \quad (3.50)$$

$$(i = 2, 3, \dots, 8; j = 2, 3, \dots, 8)$$

The system velocities and displacements are calculated from

$$\{\dot{q}_i\}^{n+1} = \{\alpha_i\} + \frac{\Delta T}{2} \{\ddot{q}_i\}^{n+1} \quad (3.51)$$

$$\{q_i\}^{n+1} = \{\beta_i\} + \frac{\Delta T^2}{6} \{\ddot{q}_i\}^{n+1} \quad (3.52)$$

$$(i = 2, 3, \dots, 8; j = 2, 3, \dots, 8)$$

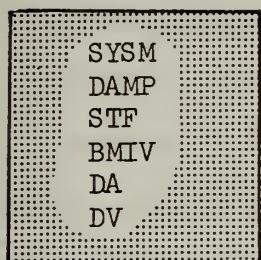
The remaining system unknowns, F_1 and F_7 may be directly calculated from equations 3.30 or 3.35 and 3.31 or 3.36. This completes the calculations for the required solution to the problem for this time step.

2. Precalculated Matrices

Returning to equation 3.32 it is noted that the mass, damping, and stiffness matrices are unique to configuration VI. Similarly from equation 3.30 and 3.31 the mass, damping, and stiffness column vectors are also unique to configuration VI. Thus these matrices need only be calculated once if stored as layers to the full system matrices. Figure 3.4 graphically indicates the matrix dimensions and elements for this and the remaining seven configurations. Additionally for the eight sets of precalculated matrices $[M]$, $[C]$, and $[K]$ the matrix variables $[\text{BMIV}]$, $[\text{DA}]$, and $[\text{DV}]$ given by equations 3.45, 3.48, and 3.49 may be precalculated for the standard ΔT and stored.

FULL SYSTEM
8 COORDINATE
MATRICES

8x8

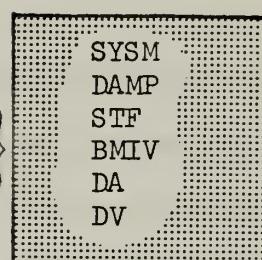


{ FIRST
EIGENVALUE
PROBLEM }

CONFIGURATION I

REDUCED SYSTEM
7 COORDINATE
MATRICES

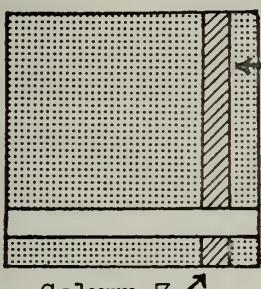
7x7



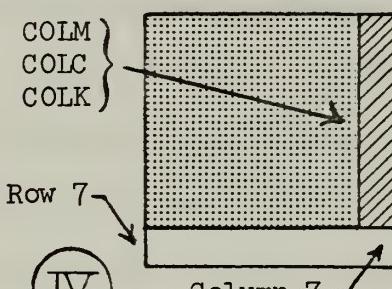
{ SECOND
EIGENVALUE
PROBLEM }

CONFIGURATION III

7x7

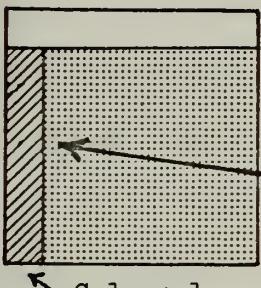


CONFIGURATION II

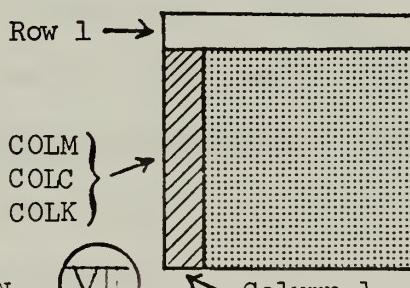


CONFIGURATION IV

7x7

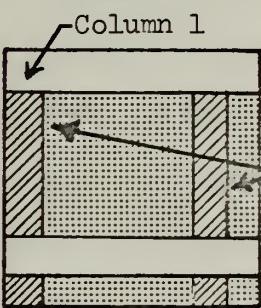


CONFIGURATION V



CONFIGURATION VI

6x6



CONFIGURATION VI

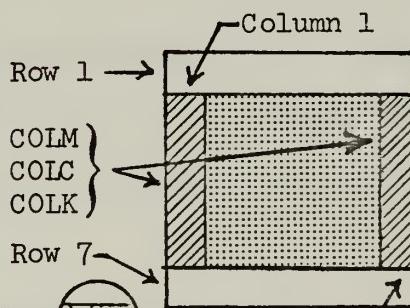


Figure 3.4 MATRIX LAYERS

F. REFINEMENTS IN THE SOLUTION

1. Incremental Time

As seen in section 3.E, equation 3.27, the normal time step is established as 1×10^{-6} seconds. Generally the system configuration will change in a time step interval, rather than at either of the end points. When this occurs the present negative monitored value calculated at $t = t^{n+1}$ and the past stored monitored value that was calculated at $t = t^n$ are used for a linear interpolation to obtain the zero point, i.e. zero force or zero displacement difference. The ΔT , now designated as $\Delta T'$, corresponding to the zero point is used to calculate the system values based on the old configurations. If the system has more than one change in configurations for a given time step, the minimum $\Delta T'$ is used for the calculations, $\Delta T'_{\min}$. The newly calculated critical value is then monitored, and if still negative, the time is again reduced by a small amount until a positive value of the monitored parameter is obtained. Figure 3.5 summarizes the above procedures for a negative force parameter.

2. Impact Forces

Anderson, in Chapter IV (Ref. 2), puts forth a good account of the study of impact forces. Numerous references are cited. Thus the historical background of impact forces in relation to this study will not be discussed here.

When the displacement difference values reduce to zero (or negative), contact at a clearance exists. This produces a step input to the velocity function and an impulse to the acceleration

function. Finite time steps are utilized to obtain solutions. Therefore one time step is sufficient to reach the step input value for the velocity curve, but two time intervals are required to account for the acceleration caused by the impulse, Figure 3.6.

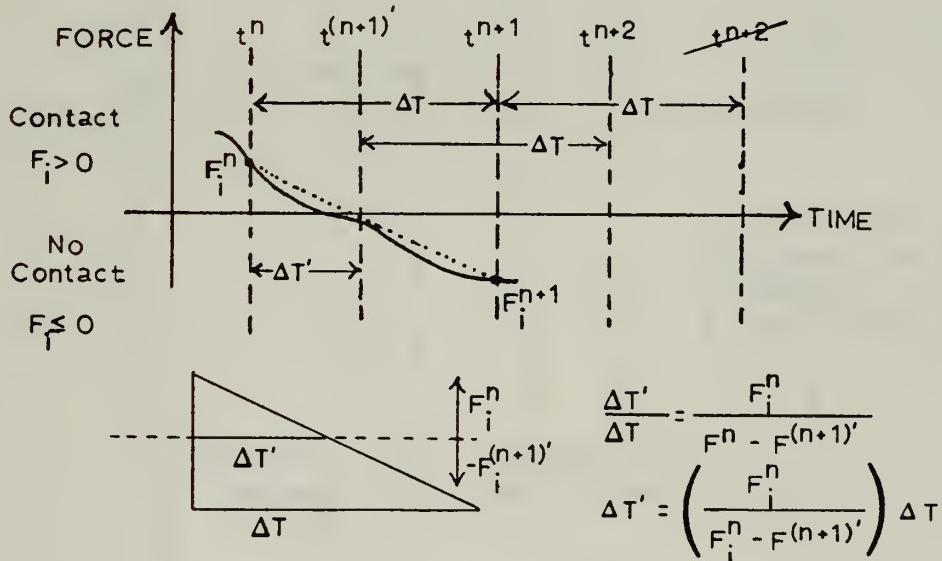
Thus the acceleration at time t^{n+1} is

$$\ddot{q}_i^{n+1} = (\dot{q}_i^{n+1} - \dot{q}_i^n) / \Delta T \quad (i = 1, 2, 7) \quad (3.53)$$

This acceleration is readily calculated for impact at the cam or valve seat ($i = 1$ or 7). However for contact at the rocker, neither \ddot{q}_2^{n+1} or \dot{q}_2^{n+1} is known. In order to solve the equation the assumption is made that

$$\dot{q}_2^{n+1} = \text{SCALE} * \dot{q}_2^n \quad (3.54)$$

The value for SCALE is arbitrarily chosen from 1.0 to 2.0, until a good system response results. Further study is suggested in this area.



then $\omega t^{(n+1)'} = \omega t^n + \omega \Delta T'$
is used to calculate system values.

Figure 3.5 Incremental Time $\Delta T'$

{ NO CONTACT AT CAM } { IMPACT & CONTACT AT CAM }

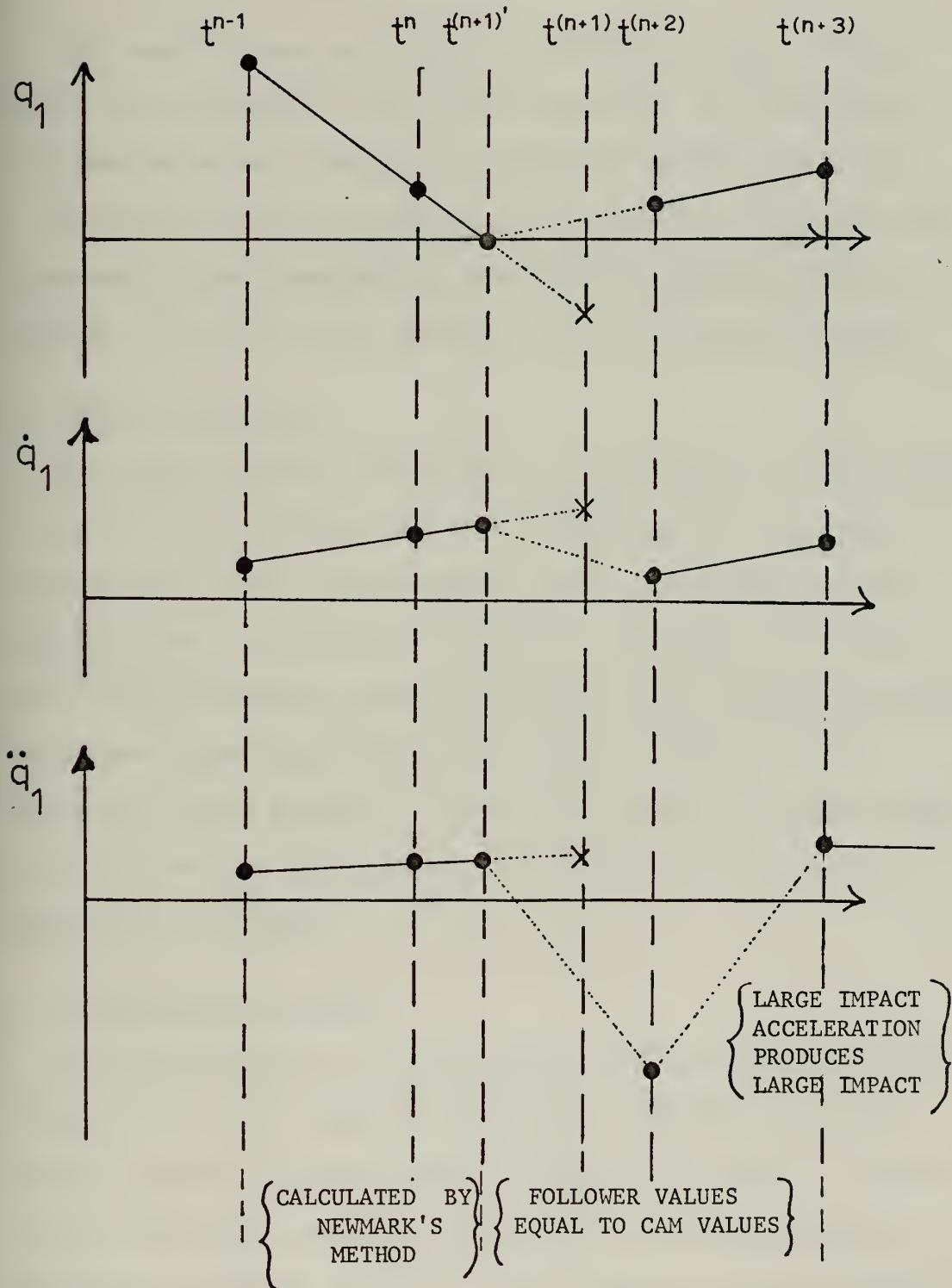


Figure 3.6 System Curves For Impact Considerations

IV. IBM-360 COMPUTER PROGRAM

This computer program is coded in FORTRAN IV, (Refs. 10-13), to run on an IBM-360/67, with OS/360 release 18. The simulation is modeled after basic concepts put forth by Anderson, (Ref. 2).

The original model program has been completely restructured and formulated to yield general procedures for the analysis of the dynamics of an elastic link mechanism with an internal clearance.

A. GENERAL DESCRIPTION

The program requires 234 K bytes of core storage for 850 degrees of cam rotation, including the CALCOMP graph buffer. The offline printer array (GRID) is not included in this core storage figure. Execution time is a function of the speed of rotation of the cam. For 11,000 RPM with 850 degrees, execution time is about 4.3 minutes. The primary output parameters fill arrays for SYSOUT = B and SYSOUT = C. The B queue is for transferring data, via punched cards, for use in the SDS-9300 computer. The C queue is for plotting graphs via the CALCOMP.

B. GENERALIZATION MATRICES

As noted in section 3.D, the number of system configurations is 2^n , where "n" is the number of clearances. Therefore, the most general computer programming was utilized, where possible. Several arrays of prestored computer "instructions" are either computer generated or formed by the user. These arrays set looping variables,

dynamically dimension work arrays, and establish program calculation flow indicators.

1. Matrix CONFIG

The 8 by 3 matrix CONFIG, Figure 4.1 was described in section 3.D. This integer array indicates the physical configurations that the model has taken based on the value of the monitored parameters. The matrix is formed by the program. The user must supply the value for the maximum number of configurations the system is permitted to take.

0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Figure 4.1 Matrix CONFIG

2. Matrix IDM

The 8 by 1 column vector IDM indicates the dimension of the matrices that are used for calculations. This array is indicated in Figure 4.2 for the three clearance system. The program parameter ISYS is the matrix dimension of the equivalent system mass, stiffness, and damping matrices and for the calculated program matrix variables BMIV, DA, and DV. The dimensions of the various layers correspond to their configuration number and to the rows of IDM, i.e. when the system is in configuration I, layer one matrices have the dimension indicated in row one of IDM. This integer array is calculated by the program.

$$\left\{ \begin{array}{c} 8 \\ 7 \\ 7 \\ 6 \\ 7 \\ 6 \\ 6 \\ 5 \end{array} \right\}$$

Figure 4.2 Matrix IDM

3. Matrix LAYER

The 8 by 1 array LAYER, Figure 4.3, is an integer column vector supplied by the user. The elements of the vector consist of ones or threes. As the name implies, the layer of the mass, stiffness, damping, BMIV, DA, and DV matrices used in the calculations correspond to the configuration number the system has taken. The eigenvalue

problem for contact at the internal clearance (layer 1) or no contact (layer 3), determines the elements of LAYER. Configurations III, IV, VII, or VIII for contact at the rocker-pushrod, use the same eigenvalue problem for the reduced system, i.e. the damping matrix is calculated with layer 3 matrices. Configurations I, II, V, or VI for no contact at the rocker-pushrod, use the full system eigenvalue problem. The damping matrix is calculated using layer 1 matrices. Figure 4.3 shows these values of LAYER for the three clearance model with two eigenvalue problems.

$$\left\{ \begin{array}{c} 1 \\ 1 \\ 3 \\ 3 \\ 1 \\ 1 \\ 3 \\ 3 \end{array} \right\}$$

Figure 4.3 Matrix LAYER

4. Matrix IQ

The 8 by 8 integer array IQ, Figure 4.4 indicates the system coordinates for which the Newmark routine will calculate the unknown quantities i.e. displacements, velocities, accelerations, and forces for a given time step. The columns of the array correspond to the configuration number except column 1 and 3. These are additional columns used with configurations VI and VIII respectively. The array is user supplied.

1	1	1	1	2	2	2	2
2	2	2	2	3	3	3	3
3	3	3	3	4	4	4	4
4	4	4	4	5	5	5	5
5	5	5	5	6	6	6	6
6	6	6	6	7	8	7	0
7	8	7	0	8	0	0	0
8	0	0	0	0	0	0	0

Figure 4.4 Matrix IQ

5. Matrix IDP

This 8 by 3 integer array, Figure 4.5, consists of zero elements except where impact may be expected to occur at the external clearances. Here again the rows correspond to the configuration number and the columns correspond to the clearance number. The matrix IDP is used in the calculation of ΔP as a portion of Newmark's method. The array is calculated by the program.

0	0	0
0	0	7
0	0	0
0	0	7
1	0	0
1	0	7
1	0	0
1	0	7

Figure 4.5 Matrix IDP

6. Matrix ICORD

The 3 by 2 integer array ICORD, Figure 4.6 indicates the coordinate pairs of the internal and external clearances. The elements are stored row wise by pairs. The rows correspond to the clearance numbers 1, 2, and 3 respectively. The first column corresponds to the index of the system coordinate that has the greater positive displacement when no contact exists at the clearance in question. The second column is assigned to the remaining index of the clearance pair. The array ICORD is user generated. Figure 4.6 shows the matrix for the three clearance problem defined in chapter 2.

1	9
2	8
7	10

Figure 4.6 Matrix ICORD

7. Matrix IMPACT

The 3 by 1 column vector IMPACT is program generated. The integer elements are continuously modified during program execution. The rows correspond to the clearance number pairs. Values 1, 2, or 7, of the system coordinates are stored when impact occurs for a given time step at a corresponding clearance.

C. DYNAMIC DIMENSIONING

The principle of dynamic dimensioning is widely used in advanced programming and often misunderstood. Thus a short example of its concepts as applied to this work is presented here. Reference to Figures 4.7 and 4.8 will aid in the understanding of the concepts.

Subroutine: THEESISBKSTEPDEL

DIMENSION: (8,8,8)

(8,8,1)

(8,8)

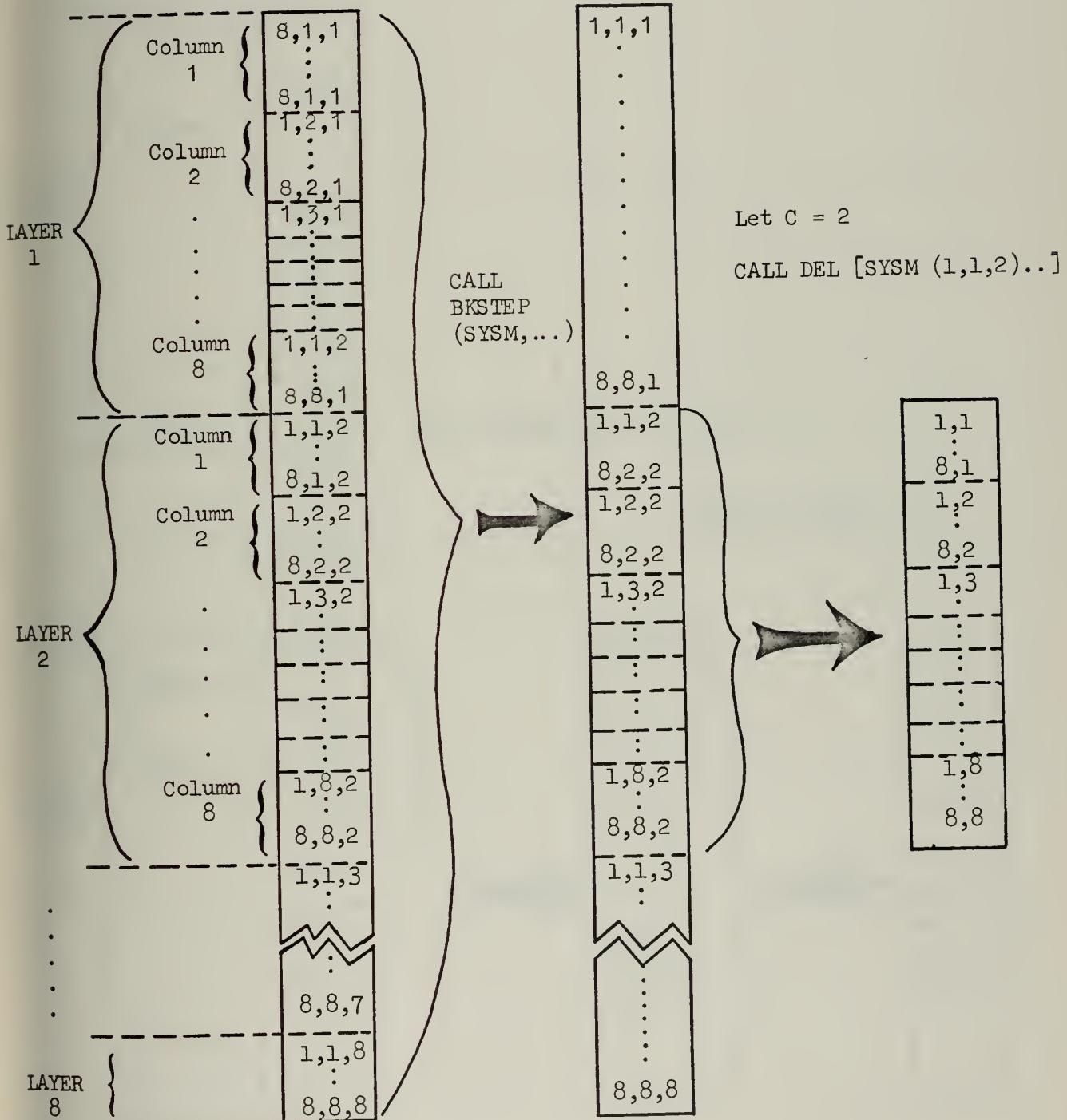


Figure 4.7 Dynamic Dimensioning of Array SYSM

ARRAY BMIV

Subroutine:

THEESIS

BKSTEP

DEL

DIMENSION:

(8,8,9)

(8,8)

(8,1)

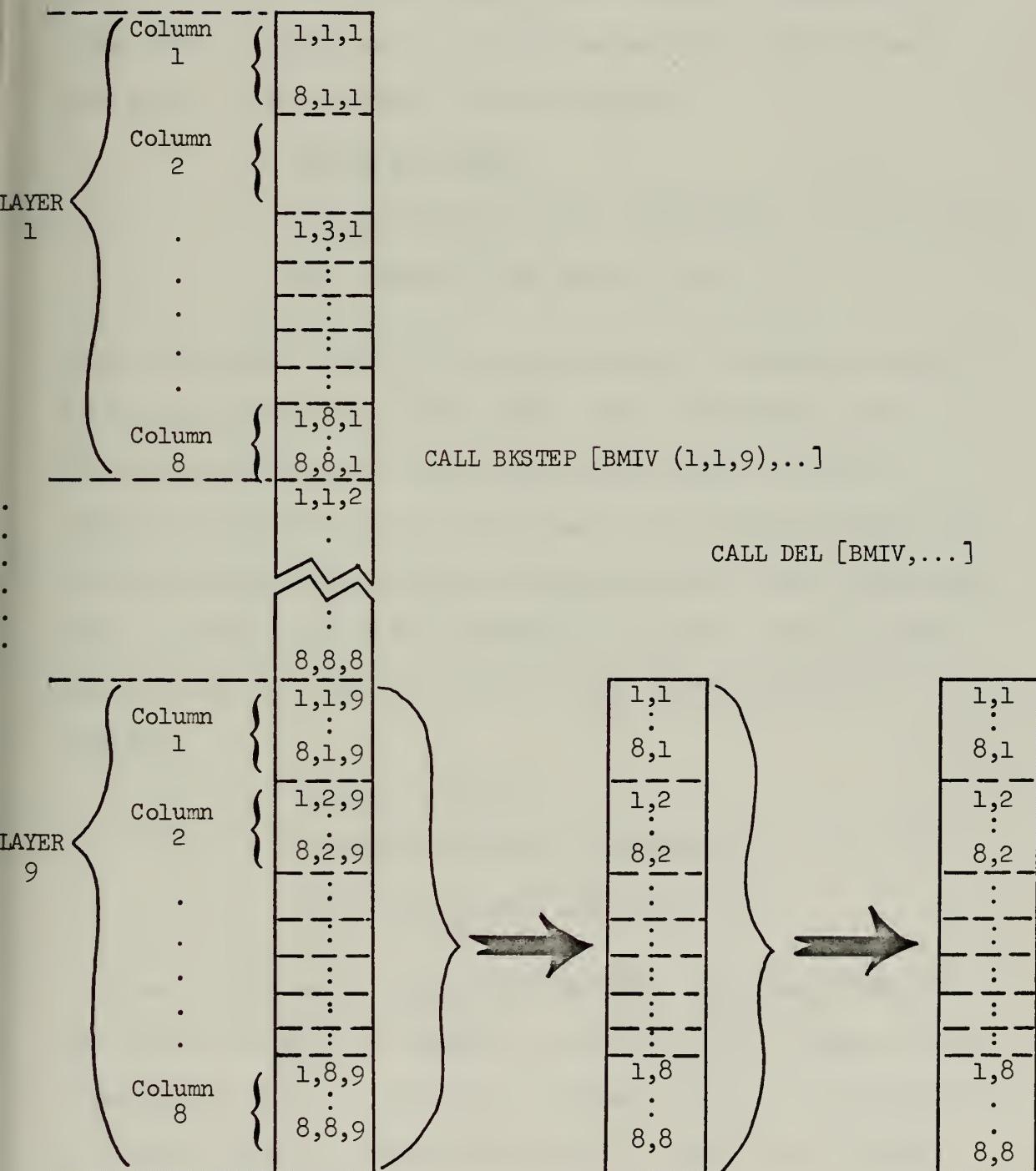


Figure 4.8 Dynamic Dimensioning of Array BMIV

In subroutine THESIS the mass array SYSM and the calculation array BMIV are exactly dimensioned indicating the total number of elements which are stored by columns in the computer. When subroutine BKSTEP is called, the entire array SYSM is passed. However the array BMIV is initialized in the call as the first value of the last layer, in this example. This is shown as:

Subroutine THESIS

```
DIMENSION SYSM (8,8,8), BMIV(8,8,9)  
CALL BKSTEP (SYSM, BMIV(1,1,9), ...)
```

Thus in subroutine BKSTEP, the array SYSM must be dimensioned with the exact number of rows and columns. Since the computer has already provided for the total space of the array, the third dimension is dynamic and if entered may be any positive number equal to or less than the originally assigned dimension. Since the entire array is utilized the third dimension is retained. For the array BMIV only the last layer is used, thus the third dimension is dropped.

Subroutine BKSTEP

```
DIMENSION SYSM(8,8,1), BMIV(8,8)  
CALL DEL (SYSM(1,1,C), BMIV, ...)
```

When subroutine DEL is called from BKSTEP the array, SYSM is now initialized with the variable C, (C = 1,2,...8). BMIV, now only considered an 8 by 8 array, is not initialized here. The dimensioning in subroutine DEL is accomplished similar to that done in BKSTEP.

The array SYSM must again be dimensioned as above with the third dimensioning parameter being dynamic. The array BMIV is used as a double dimensioned array in subroutine DEL. By arguments similar to those above, the columns of the array may now also be dynamically dimensioned as:

Subroutine DEL

DIMENSION SYSM(8,8), BMIV(8,1)

Figure 4.7 for array SYSM and Figure 4.8 for array BMIV graphically indicate the manner in which these arrays are passed from the subroutines, THESIS-BKSTEP-DEL.

D. PROGRAM STRUCTURE

The main program only sets up the plotting arrays and then calls subroutine THESIS, the main routine that controls the program execution. The calling order is presented in Figure 4.9. The CALCOMP graphing subroutines indicated by an asterisk are supplied by the computer center.

E. SUBROUTINES

The function of each of the subroutines and a description of their execution are included in this section. The concept of execution of object modules stored on disk was used to a large extent. This accounts for the large number of subroutines and their limited content. Flow graphs for the majority of the routines are collected and presented at the end of this section. Each subroutine is listed in its entirety in the IBM-360 Computer Program Listing chapter.

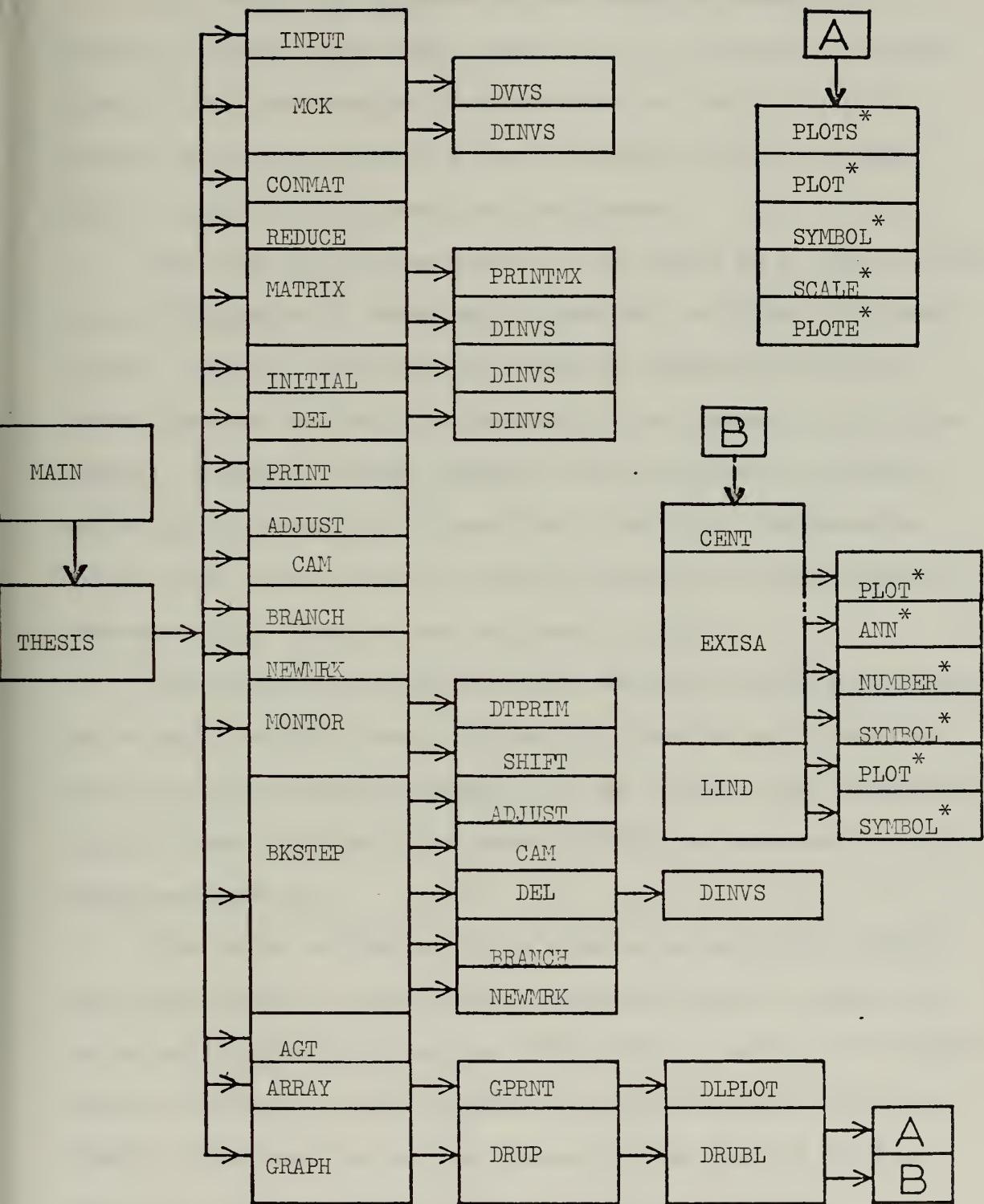


Figure 4.9 IBM-360 Computer Program Structure

1. Subroutine THESIS

This subroutine controls program execution, establishes calculation arrays (MCK, CONMAT, MATRIX, DEL), initializing parameters (INPUT, INITIAL) and contains the major time step loop, $\Delta T = 10^{-6}$ seconds. As seen in Figure 4.9, most subroutines are called from THESIS. Figure 4.10 indicates the flow diagram.

The time parameter $\omega \Delta T$ advances by an amount $\omega \Delta T$. The previous values of displacement, velocity, acceleration, and force are stored (ADJUST). The new cam values are calculated (CAM) and the known system parameter are initialized according to the present configuration (BRANCH). Newmark's method (NEWMARK) is then employed to calculate the remaining unknown system quantities. The system configuration value is next stored prior to checking the newly calculated quantities (MONTOR) for any change in the configuration number.

The system is now checked to see whether an impact has occurred and at what coordinate index. If the configuration has changed, the quantities are recalculated (BKSTEP) for the time that the configuration changed. Then force values are reset (FIXFOR) to coorespond to the system configuration.

The amount of degrees of cam rotation is calculated. If the next whole degree has been reached the system values are output for use as the graphics (AGT) and draw (ARRAY) data. Next if the required number of degrees have been evaluated the desired graphs are plotted (GRAPH). Otherwise, the program branches to the start of the time step loop (statement number 360) and the cycle is repeated.

2. Subroutine BKSTEP

This subroutine is similar to subroutine THESIS in that all program variables are recalculated to correspond to the time at which the system configuration changed. Once this routine is entered only the parameter corresponding to the configuration shift is monitored until a positive value is calculated. Figure 4.11 indicates the flow diagram.

3. Subroutine MCK, DVVS, and DINVS

This subroutine originally coded by Anderson and Winfrey, was modified to account for the additional eigenvalue problem. The sizes and material properties of the system members are assigned. The element of the full and reduced system mass and stiffness are assigned to their respective arrays, SYSM and STF. The damping ratio, (ZTA) is set on the basis of the fundamental frequency of the system being considered. The eigenvalue problems are solved by subroutine DVVS. Subroutine DINVS is called to obtain matrix inverses..

4. Newmark's Method Subroutines

Subroutines NEWMRK and DEL are the primary program calculation routines, and considered together comprise Newmark's method. The flow diagram for subroutine DEL is seen in Figure 4.12 and for subroutine NEWMRK, Figure 4.13.

(a) Subroutine DEL

Subroutine DEL calculates the eight layers of the matrix arrays BMIV, DA, and DV, based on the full time step ΔT . These matrices are precalculated and stored. When DEL is called from subroutine BKSTEP, an additional layer of the above matrices is calculated as required, with a partial time step, $\Delta T_{\min} < 10^{-6}$ seconds.

(b) Subroutine NEWMRK

Subroutine NEWMRK calculates the coordinate accelerations by Newmark's method. The theory of this method is presented in appendix B. Its relation to this study is presented in section 3.E. After the system accelerations are known, the displacements and velocities can be calculated. Finally the forces are calculated at clearance coordinates 1, 2, and 7.

5. Subroutine for Shifting Configurations

The two subroutines that provide the program the ability to shift from one configuration to another, based on the values of the monitored parameters, are MONTOR and SHIFT. Figure 4.14 indicates the flow diagram for subroutine MONTOR. The subroutine that determines the amount by which the time step should be reduced, after a change in the configuration has been indicated, is DTPRIM. The flow diagram for this subroutine is seen in Figure 4.16.

(a) Subroutine MONTOR

Subroutine MONTOR checks the forces and/or the displacement differences at the three clearances to determine if any has become negative. As indicated by the shifting flow diagram, Figure 2.6, only three quantities must be monitored at each clearance, either forces and/or displacements differences. When at least one of the values becomes negative, subroutine DTPRIM is called.

(b) Subroutine DTPRIM

Subroutine DTPRIM linearly interpolates for a best estimate of the time the monitored value became negative, i.e. when the system changed to a different configuration. The calculated time

is stored in array DTP. If more than one change to the system configuration occurs in a single time step, the first change is chosen, i.e. the minimum value among the elements of the array DTP. The clearance index number that causes the system to change is then sent to subroutine SHIFT.

(c) Subroutine SHIFT

Subroutine SHIFT, described in section 3.D returns the new configuration based on the parameters sent by MONTOR.

6. Subroutine INPUT

This subroutine reads as data the formal parameters that are frequently varied. Figure 4.17 presents the flow diagram for this subroutine. These parameters include revolutions per minute (RPM) of the cam, clearance indicators (CLEAR, PIN 2&8) printing indicators (MPRINT, WTDEG, KPRINT, NCOUNT, IPRINT), output data indicators (IDRAW, AGTOUT, CARDS, TAPE IPLOT) and the time step (DELT). Graph titles are read into the arrays XNAME, YNAME, LABEL, and TITLE. The array TITLE is modified to indicate programs without a valve seat. The array LABEL is modified to indicate a two clearance problem. The lengths of the member elements are assigned (REALTH, AGTLTH) and the lever arm of the rocker (ROCKER) is calculated.

7. Subroutine CONMAT

This subroutine establishes two of the program generalization arrays, CONFIG and IDP. Figure 4.18 shows the flow diagram. The routine is called only once during the program execution. The coding for both arrays may be easily expanded to provide for problems with more than three clearances.

8. Subroutine INITIAL

This subroutine initiates the starting values of displacement, velocity, acceleration, and force for the system coordinates. The starting configuration VII, requires that layer seven matrices be used for all calculations. Subroutine DINVS is called to establish the inverse stiffness matrix necessary for the displacement calculations. The program variables for the follower are set equal to the cam values and those for the left rocker arm are set equal to the pushrod values by calling subroutine ADJUST.

9. Subroutine CAM

This short subroutine calculates the displacement, velocity, and acceleration at the cam (coordinate 9) for the given wt. The flow diagram is presented as Figure 4.15. The displacement is given by the equation:

$$q_9 = .5 \sin \omega t + XZERO$$

This routine is called at least once, every time increment from subroutine THESIS. Additionally cam is called by subroutine BKSTEP for intermediate values of wt.

10. Subroutine BRANCH

This subroutine is called prior to making calculations for a new time step. Figure 4.19 indicates the flow diagram. The eight sections of the routine initialize the system parameters corresponding to its present configuration. Additionally the dimension parameter ISYS is set from the array IDIM. A correction of the above established acceleration is accomplished when the array IMPACT indicates that a

clearance displacement difference is zero. The adjustment of the acceleration produces a corresponding increase in the later calculated force.

11. Subroutine ADJUST

This short subroutine equivalences the displacement, velocity, and acceleration quantities at a system coordinate to those at another system coordinate. For example, when contact exists between cam and follower the above named quantities for coordinate one are set equal to those for coordinate nine. Figure 4.20 presents the flow diagram.

12. Subroutine FIXFOR

This subroutine is primarily used to zero the calculated forces on the basis of the system configuration. This is necessary to eliminate small negative or small positive forces from being plotted. This routine has no effect on the program calculations. Its flow diagram is seen as Figure 4.21.

13. Output Subroutines

(a) Subroutine AGT

The subroutine AGT outputs data on punched cards or magnetic tape for use in the SDS-9300 program calculations of the graphic display. The flow diagram is indicated in Figure 4.22. This subroutine first adjusts the system displacement coordinates by an amount XZERO so as to present the initial position of the graphics display as horizontal. In addition to the displacement coordinates, reference values are sent for the cam rotation and the valve seat position.

(b) Subroutines ARRAY, GRAPH, GPRINT, OL PLOT

The subroutine ARRAY is called for the first value of each whole degree and establishes the array FPLOT, which is used for the plotting of graphs on the CALCOMP or offline printer. Figure 4.23 presents the flow diagram. Subroutine GRAPH is the last subroutine called by THESIS after the desired number of degrees has been evaluated. This subroutine, as seen in the flow diagram, Figure 4.24, organizes the calls to the offline printer subroutine OL PLOT for IPLOT equal to YES and/or the CALCOMP subroutine GPRINT for IDRAW equal to YES. The flow diagram for subroutine GPRINT is indicated in Figure 4.25. Subroutine OL PLOT is a modified version of the Computer Center provided routine UT PLOT. One of the modifications provides for an expansion of the graph output to a maximum of six pages. The number of pages desired is assigned to the program parameter, IP, by the user. Additionally the users must adjust the dimensions of the main program array, GRID to conform with the desired pages. GRID must be dimensioned as LINES by 81 where the dimension parameter LINES is:

$$\text{LINES} = (80)(\text{IP}) + 1$$

(c) Subroutine DRU--

The many subroutines beginning with DRU-- were written for plotting graphs on the CALCOMP plotter by Winfrey. This package also contains computer center provided routines, indicated by an asterick in Figure 4.9.

(d) Subroutine LIST (DEBUG)

Subroutine LIST outputs the numerical formal parameters on the offline printer for either "debugging" purposes or as reference values when viewing the graphs/graphics.

(e) Subroutine PRNTMX

Subroutine PRNTMX outputs all the system matrices,
when the program parameter MPRINT is equal to YES.

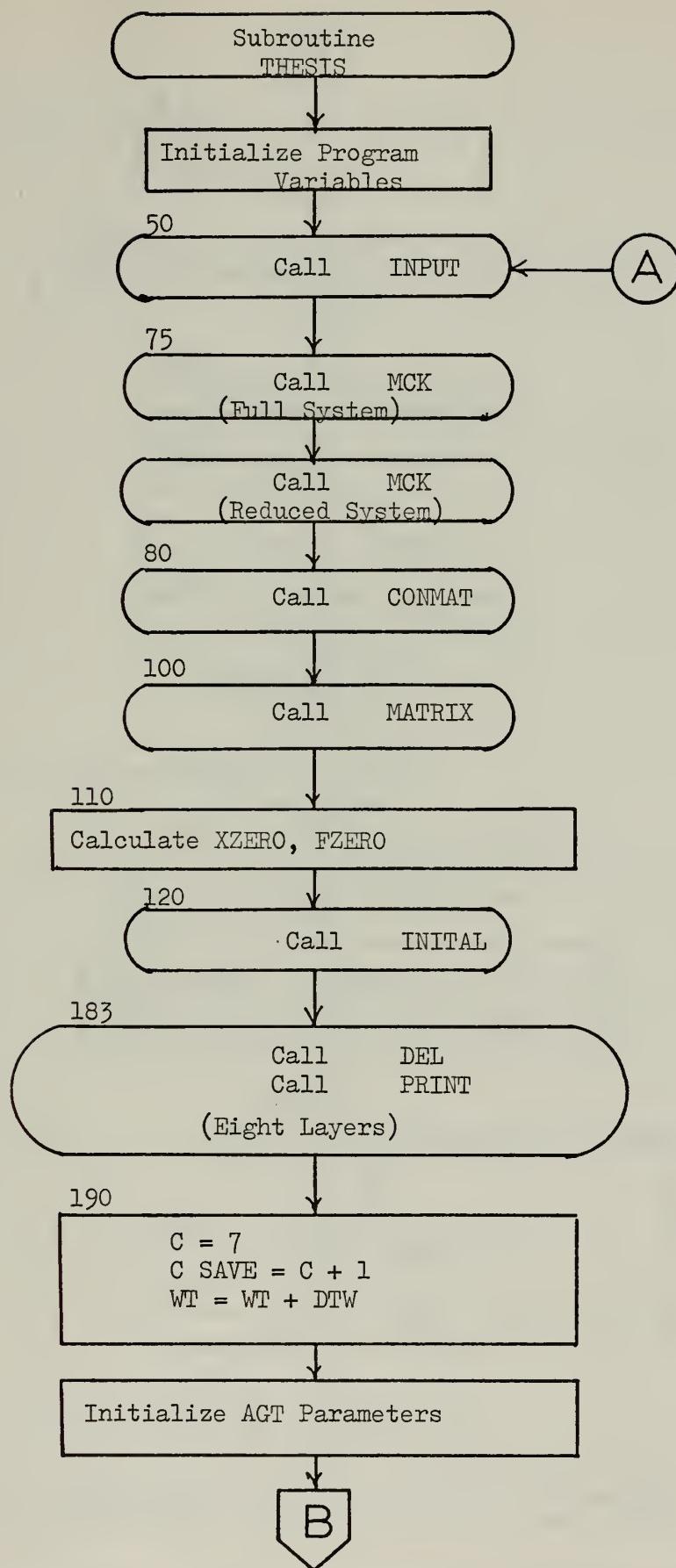


Figure 4.10 Subroutine THESIS Flow Diagram

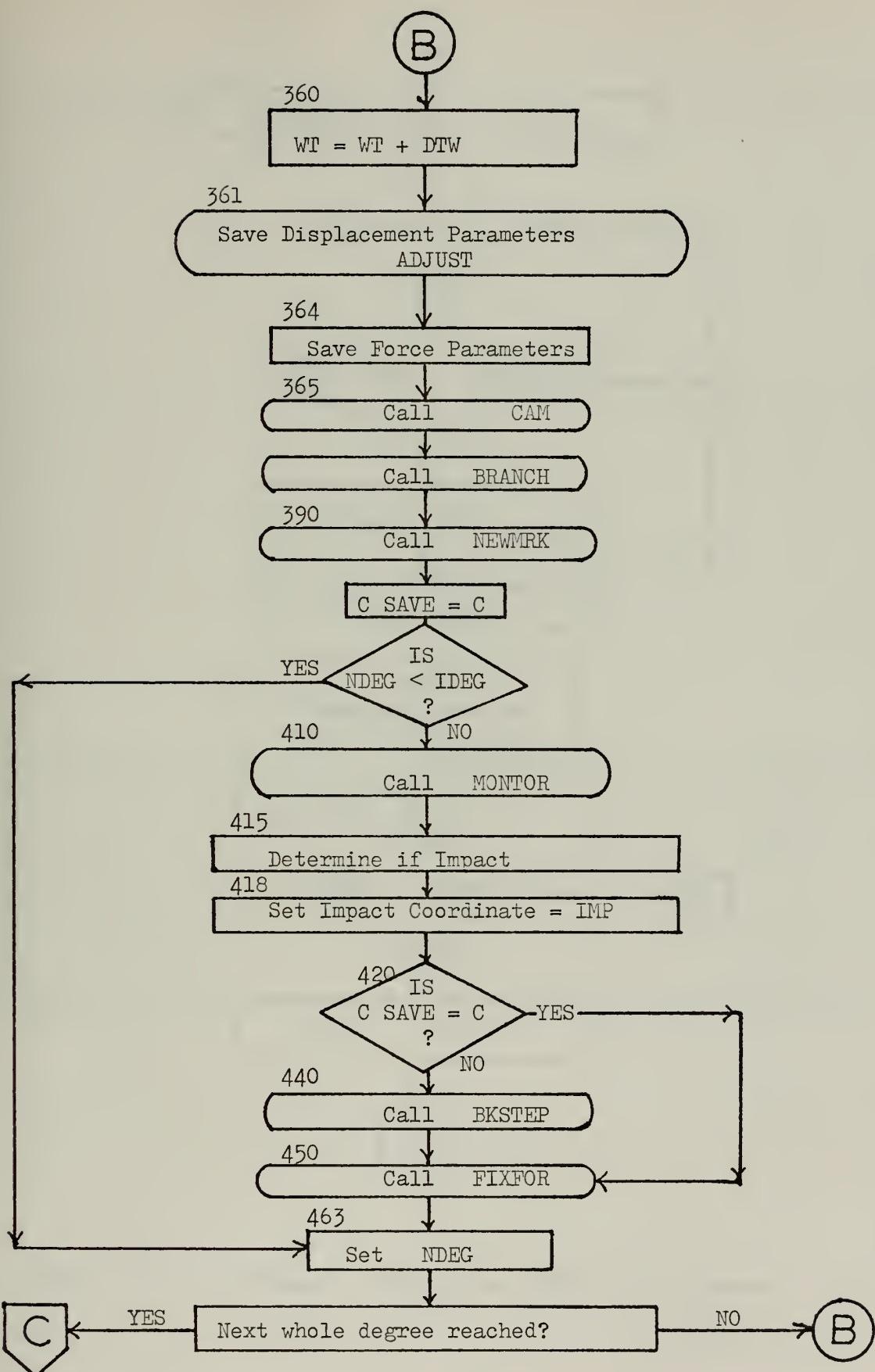


Figure 4.10 Continued - Subroutine THESIS Flow Diagram

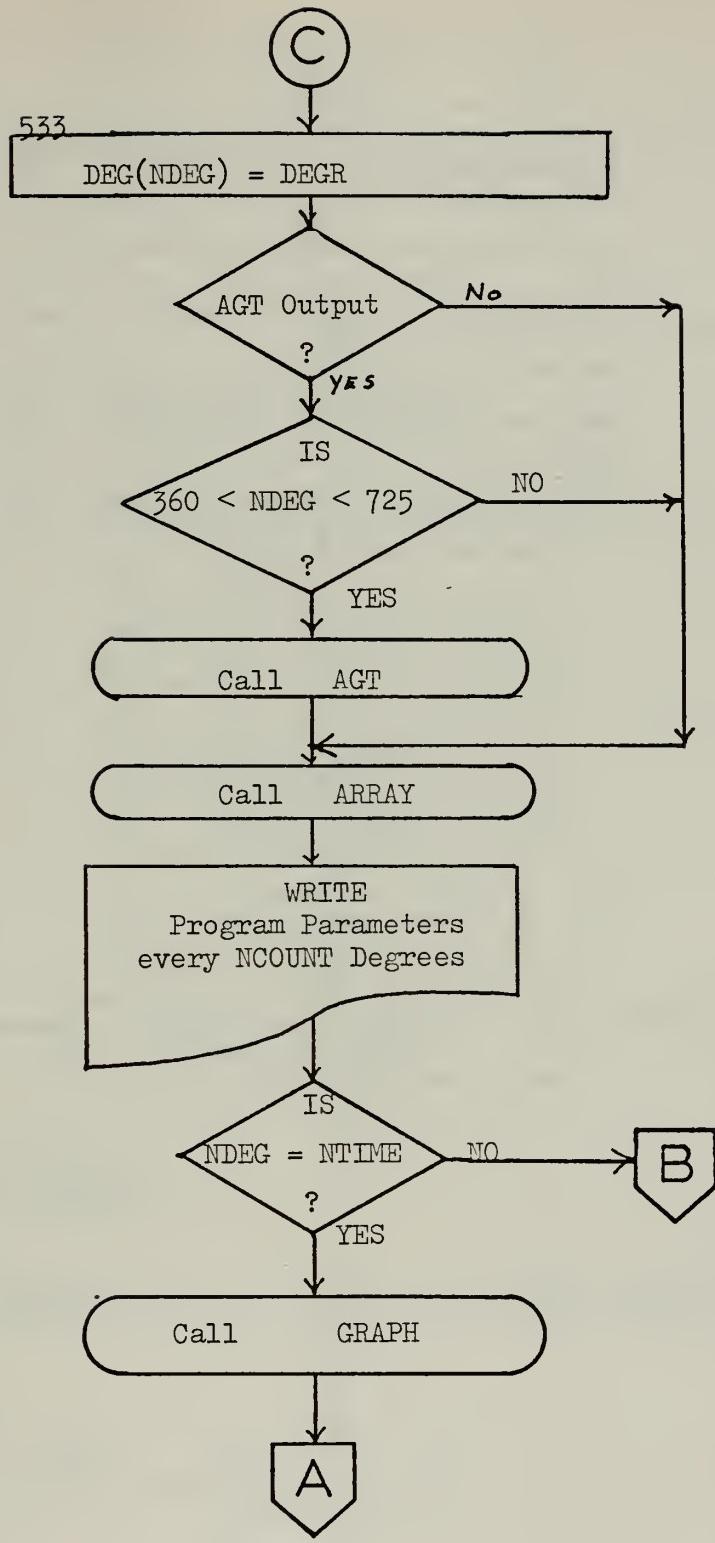


Figure 4.10 Continued - Subroutine THESIS Flow Diagram

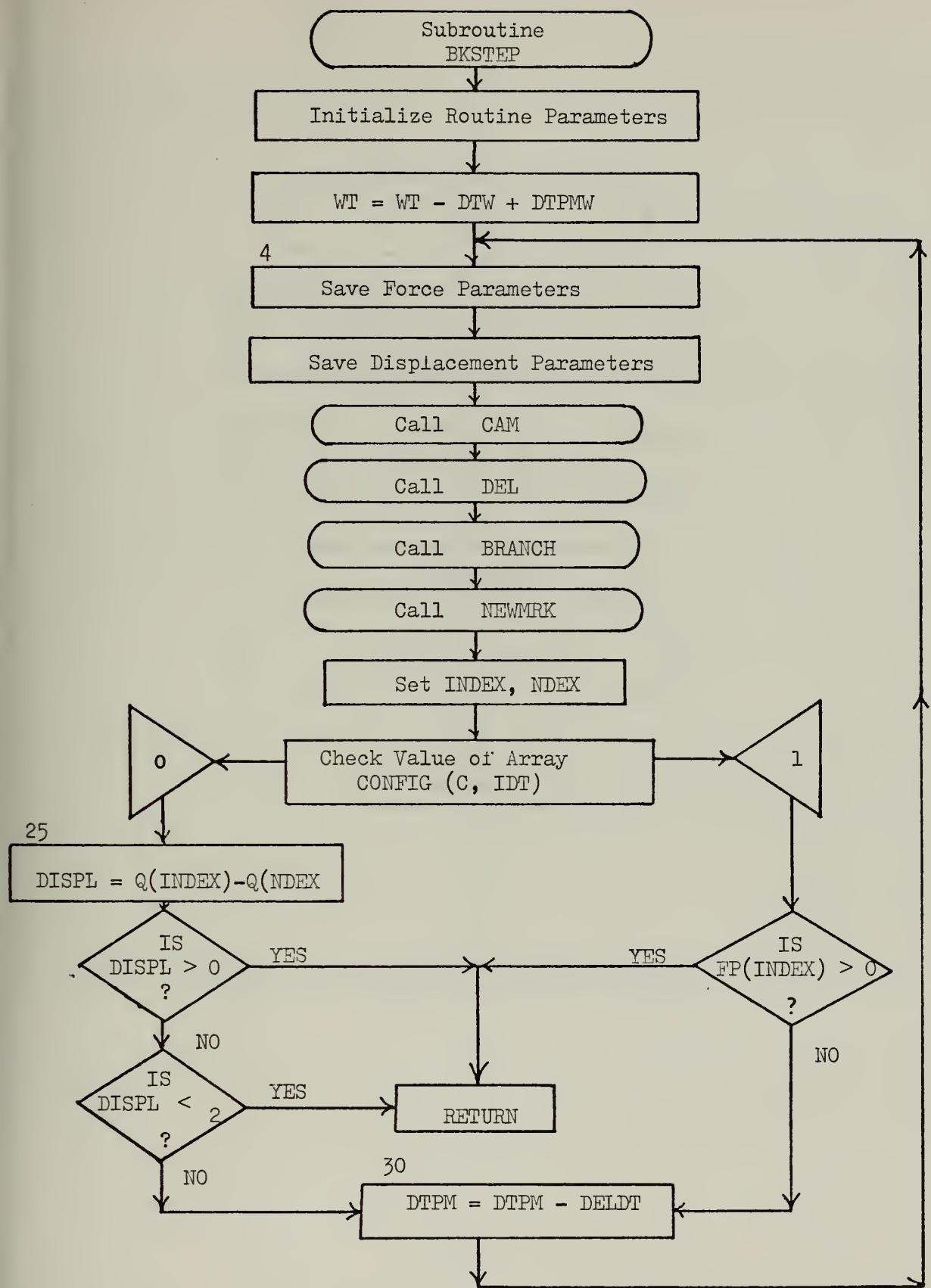


Figure 4.11 Subroutine BKSTEP Flow Diagram

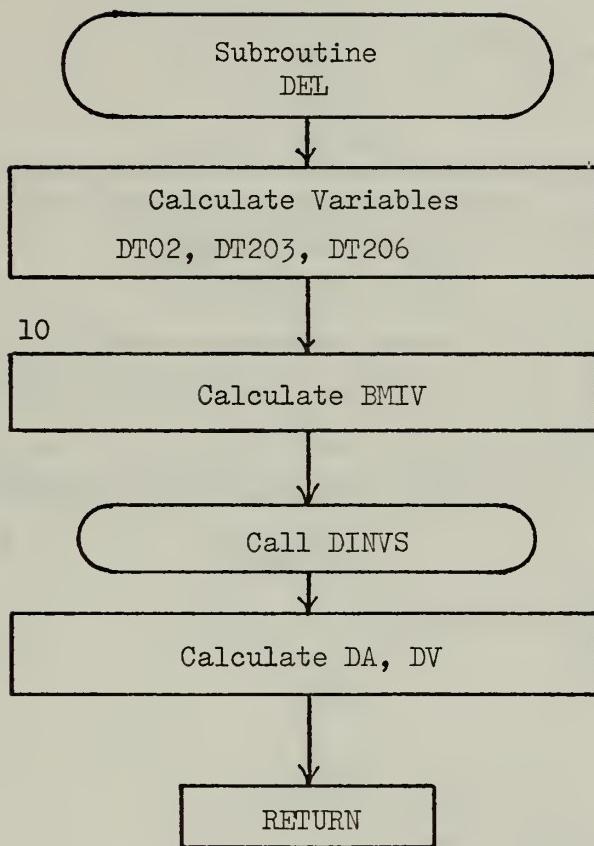


Figure 4.12 Subroutine DEL Flow Diagram

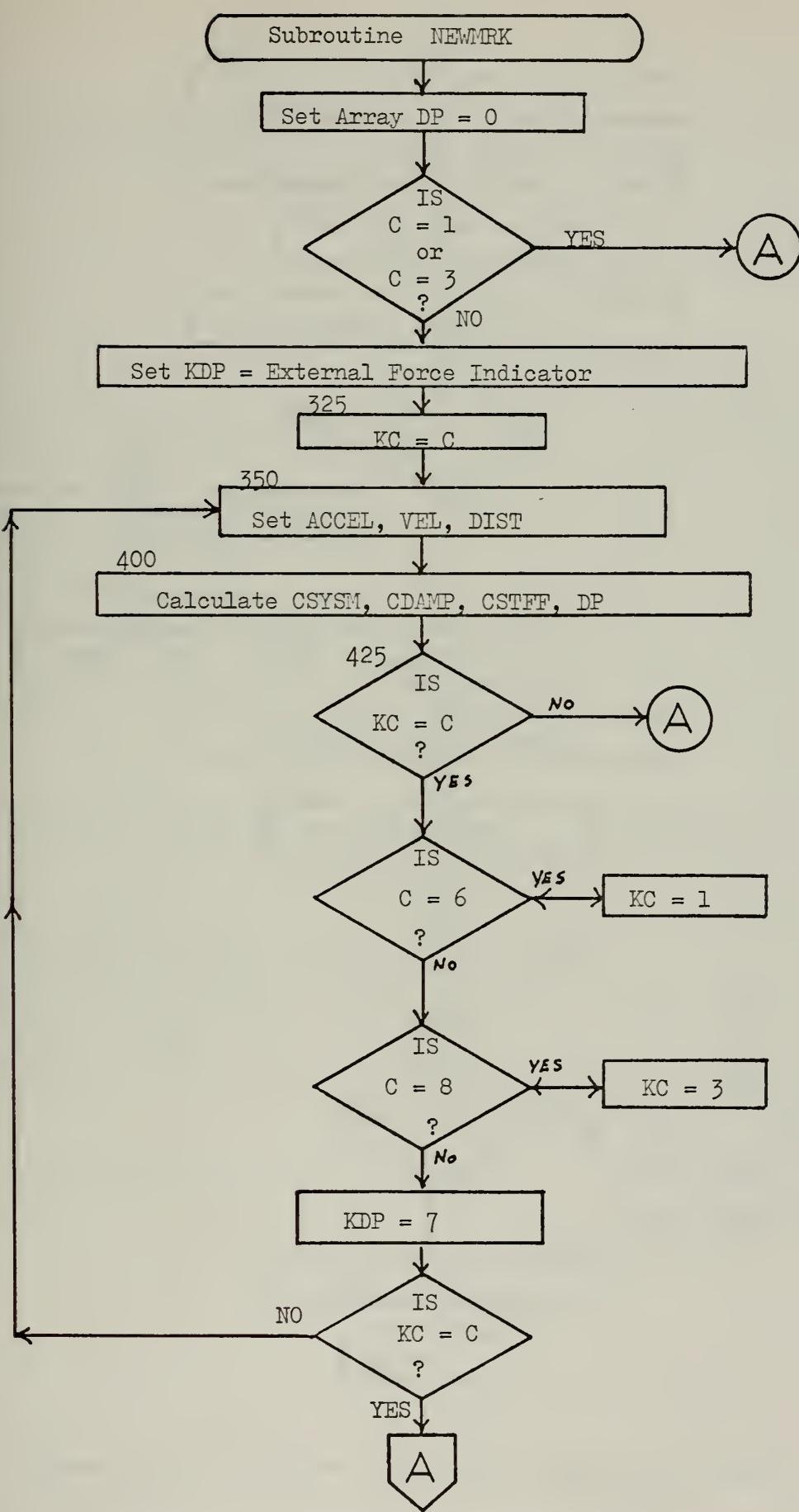


Figure 4.13-1 Subroutine NEWMRK Flow Diagram

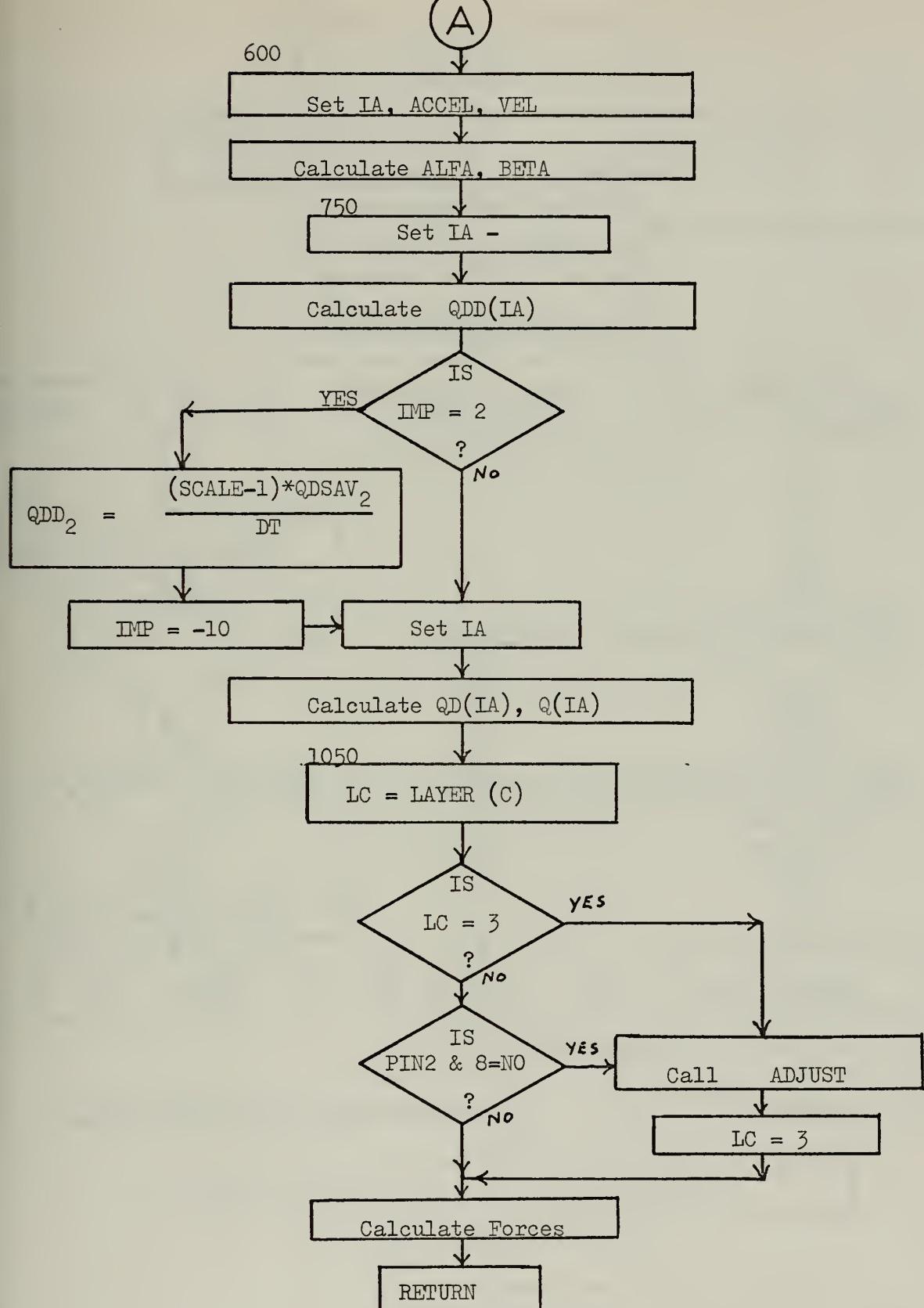


Figure 4.13-2 Continued - Subroutine NEWMRK Flow Diagram

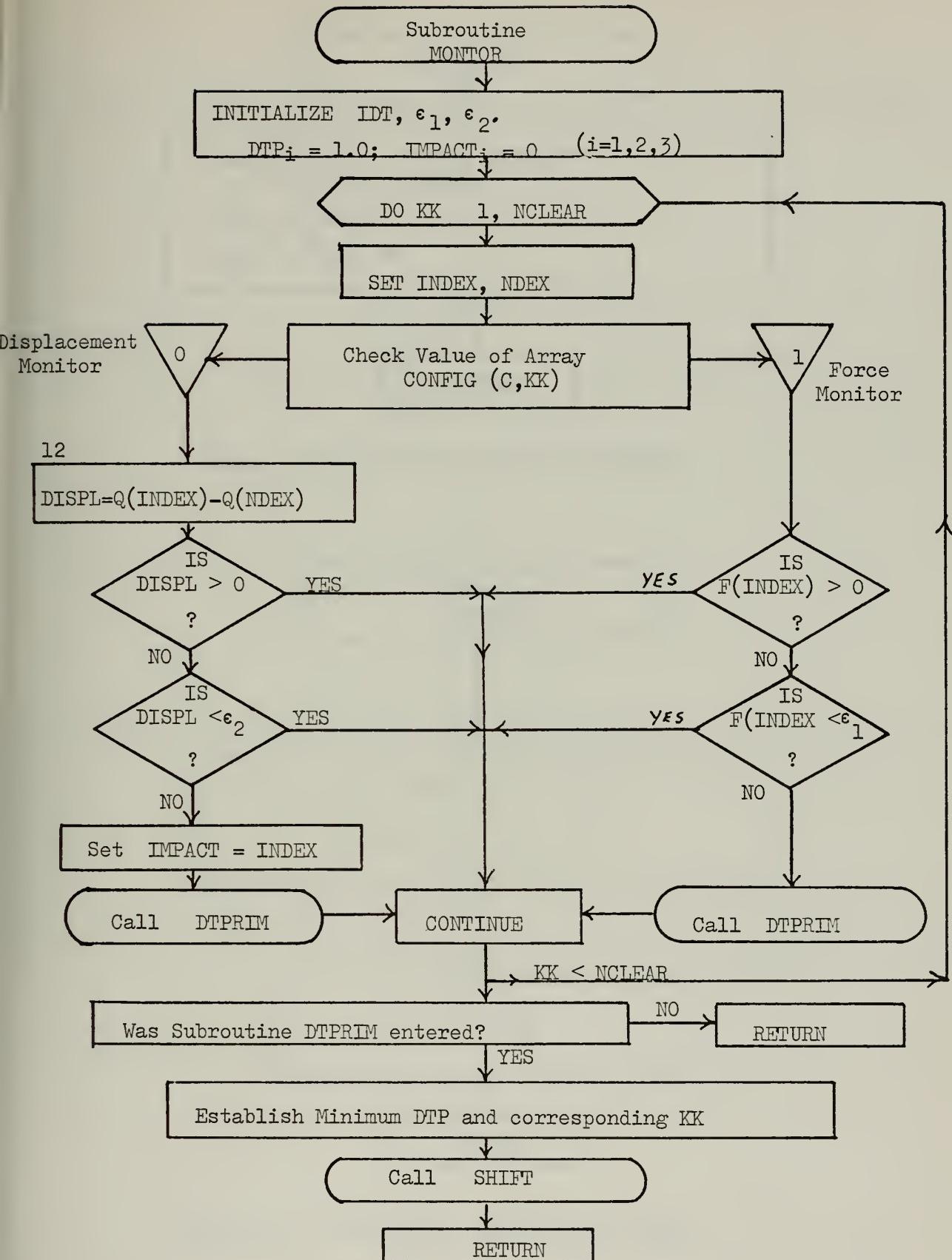


Figure 4.14 Subroutine MONTOR Flow Diagram

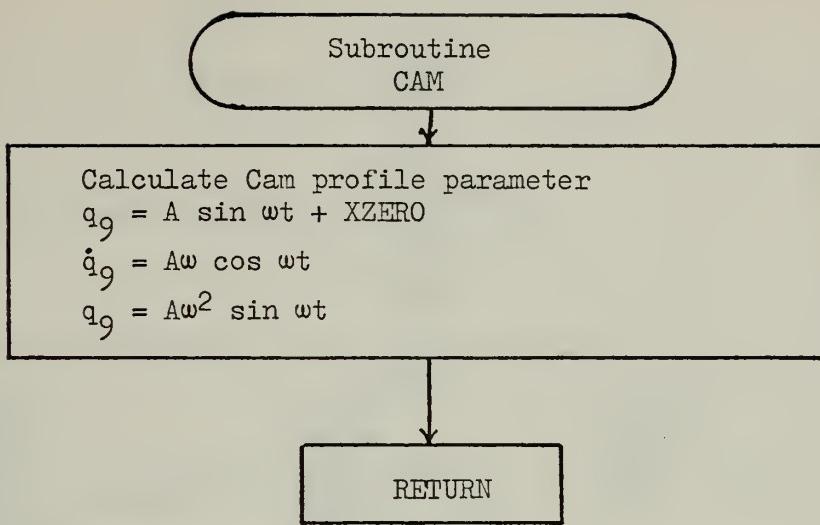


Figure 4.15 Subroutine CAM Flow Diagram

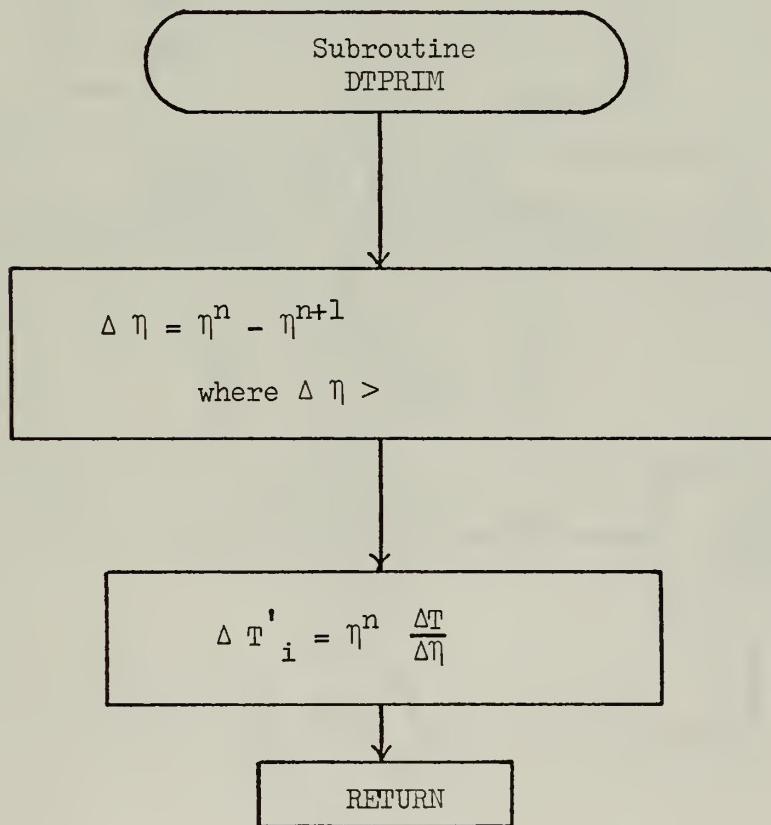


Figure 4.16 Subroutine DTPRIM Flow Diagram

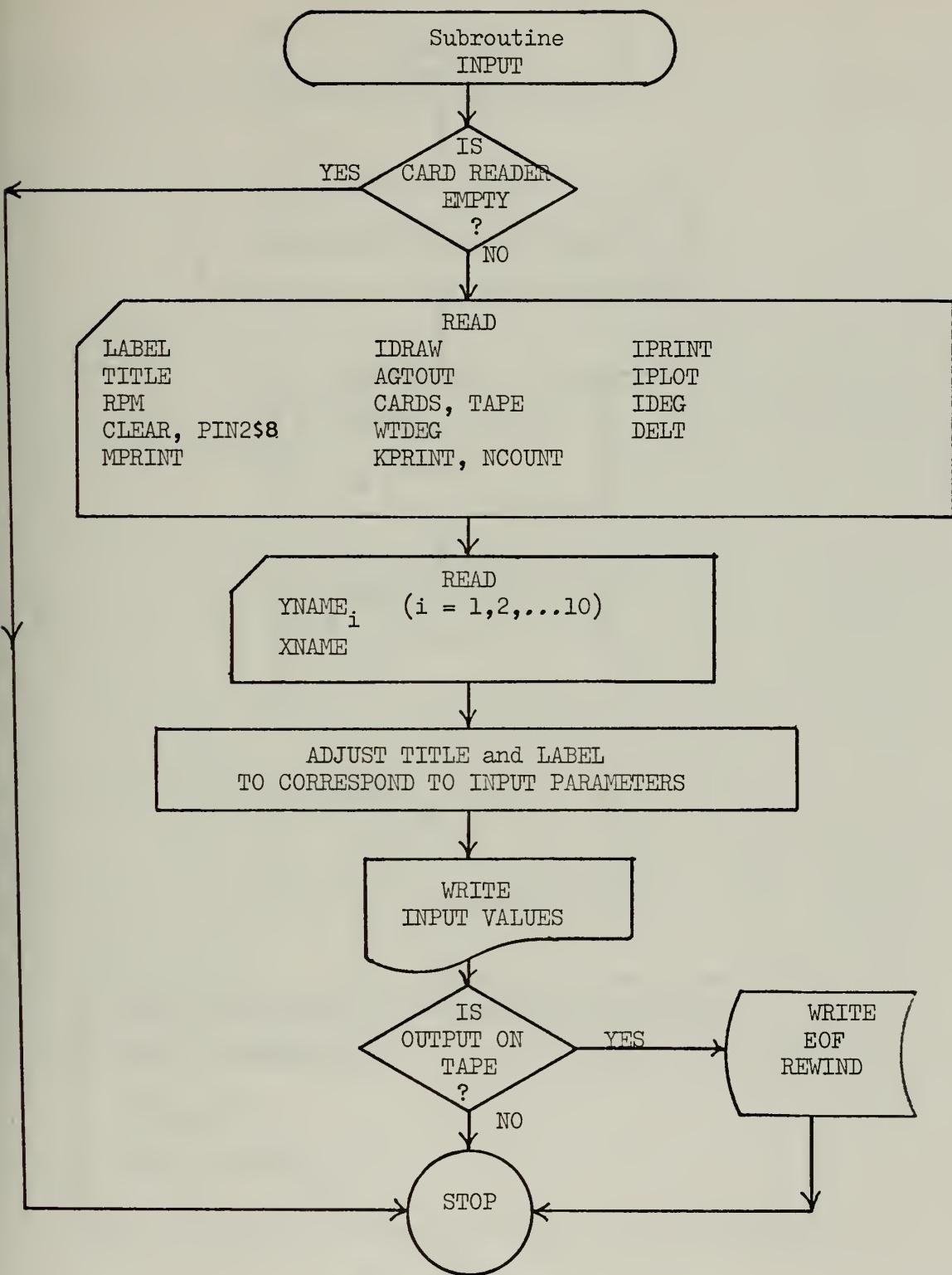


Figure 4.17 Subroutine INPUT Flow Diagram

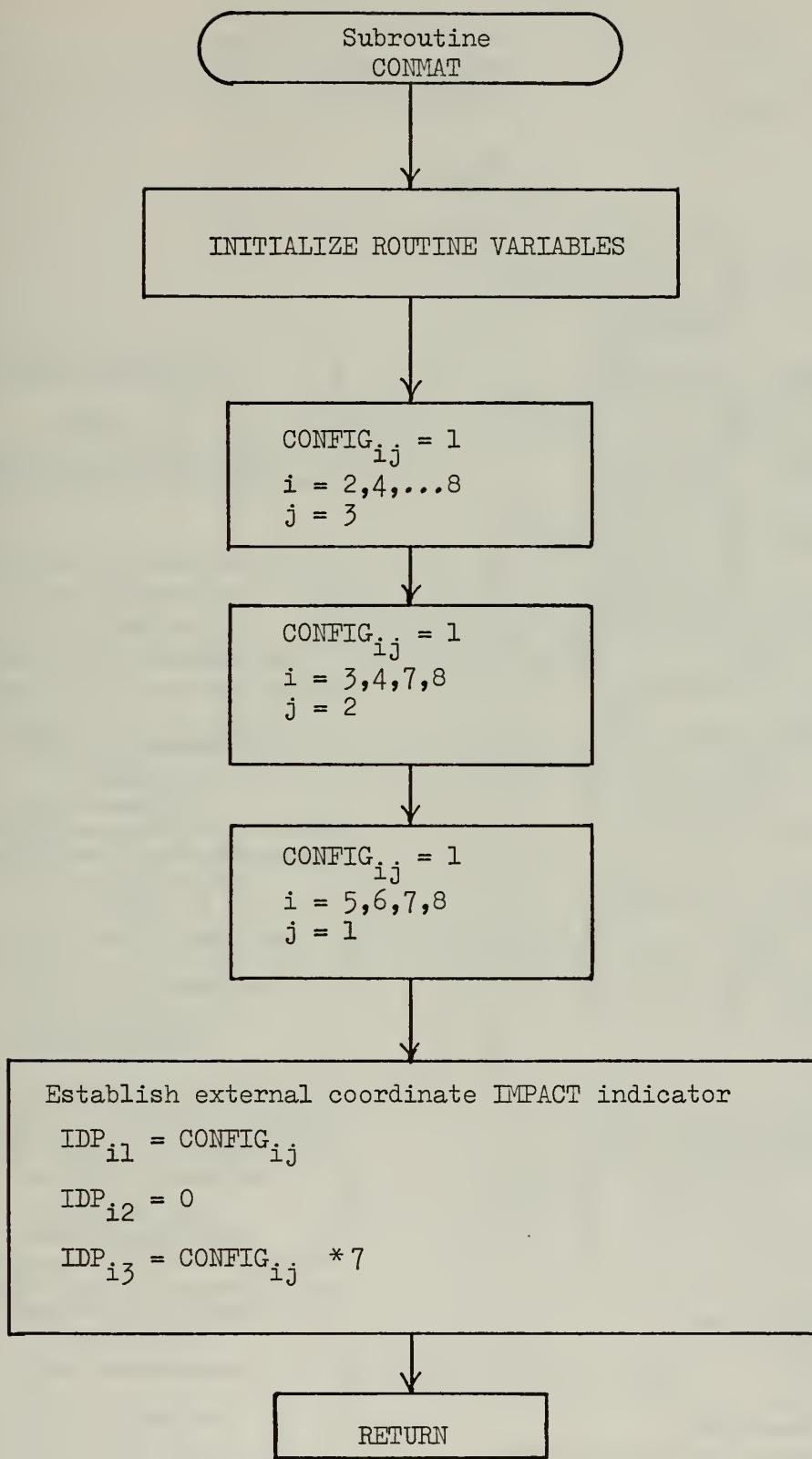


Figure 4.18 Subroutine CONMAT Flow Diagram

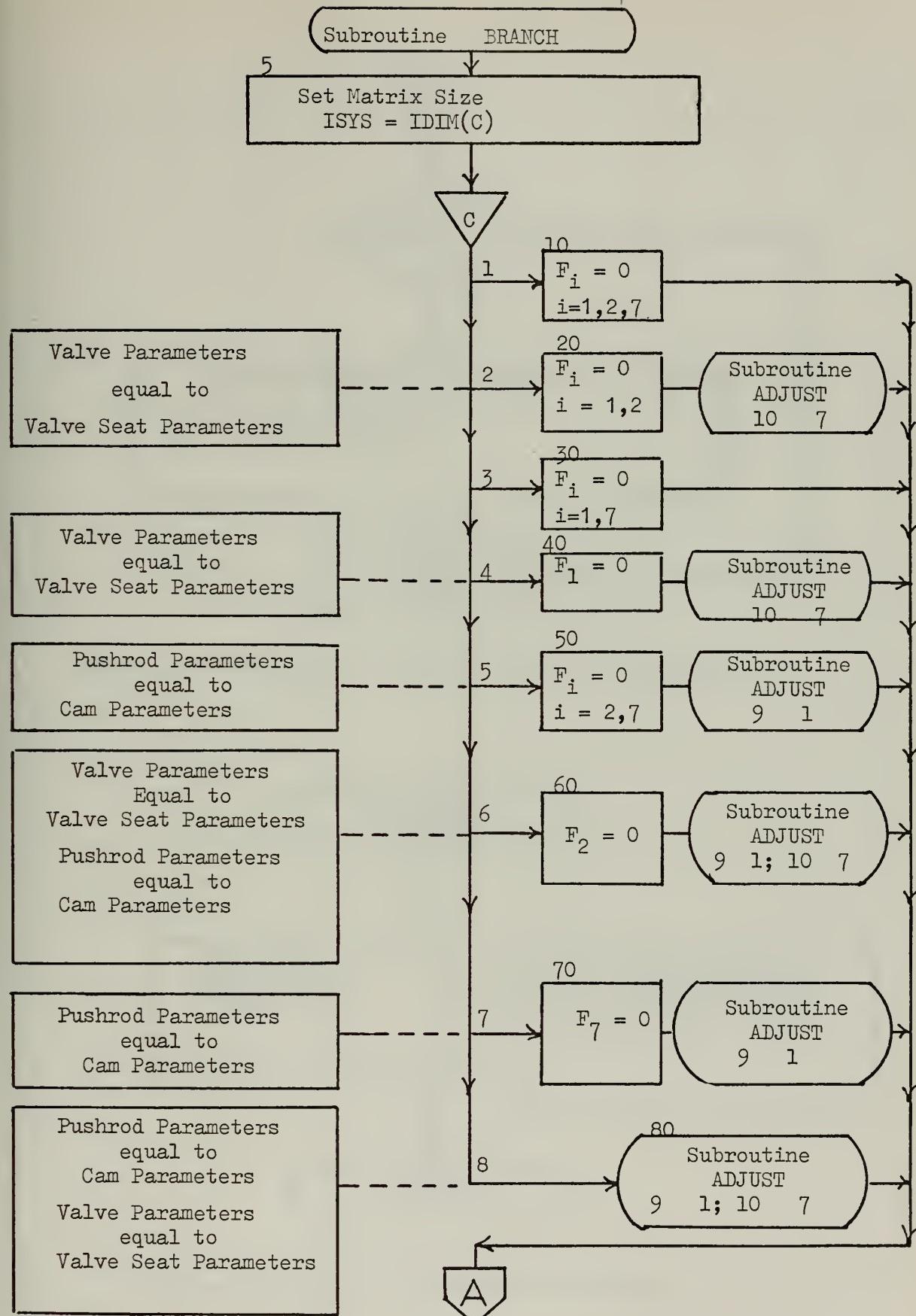


Figure 4.19 Subroutine BRANCH Flow Diagram

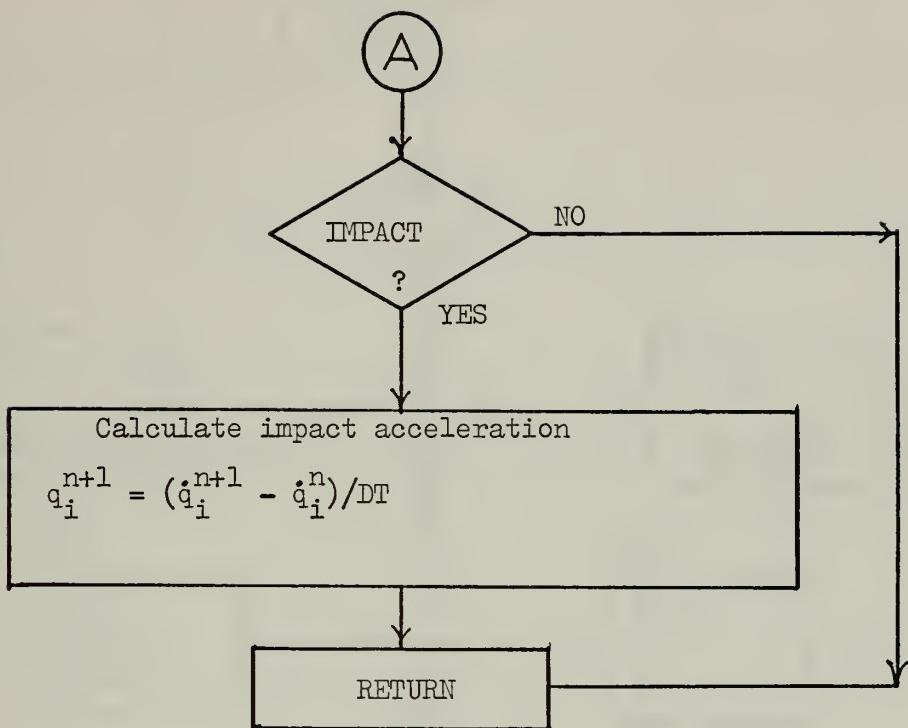


Figure 4.19 Continued - Subroutine BRANCH Flow Diagram

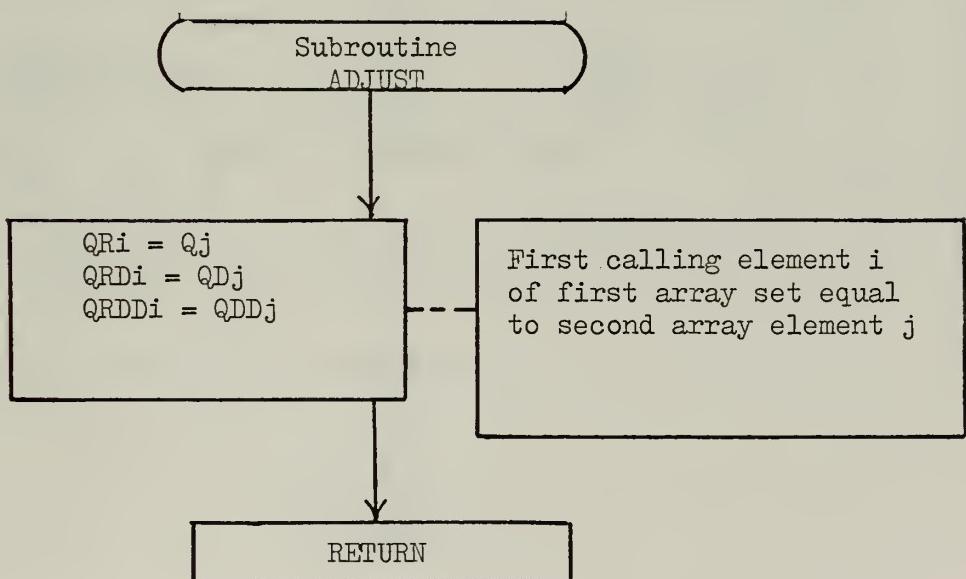


Figure 4.20 Subroutine ADJUST Flow Diagram

Subroutine
FIXFOR

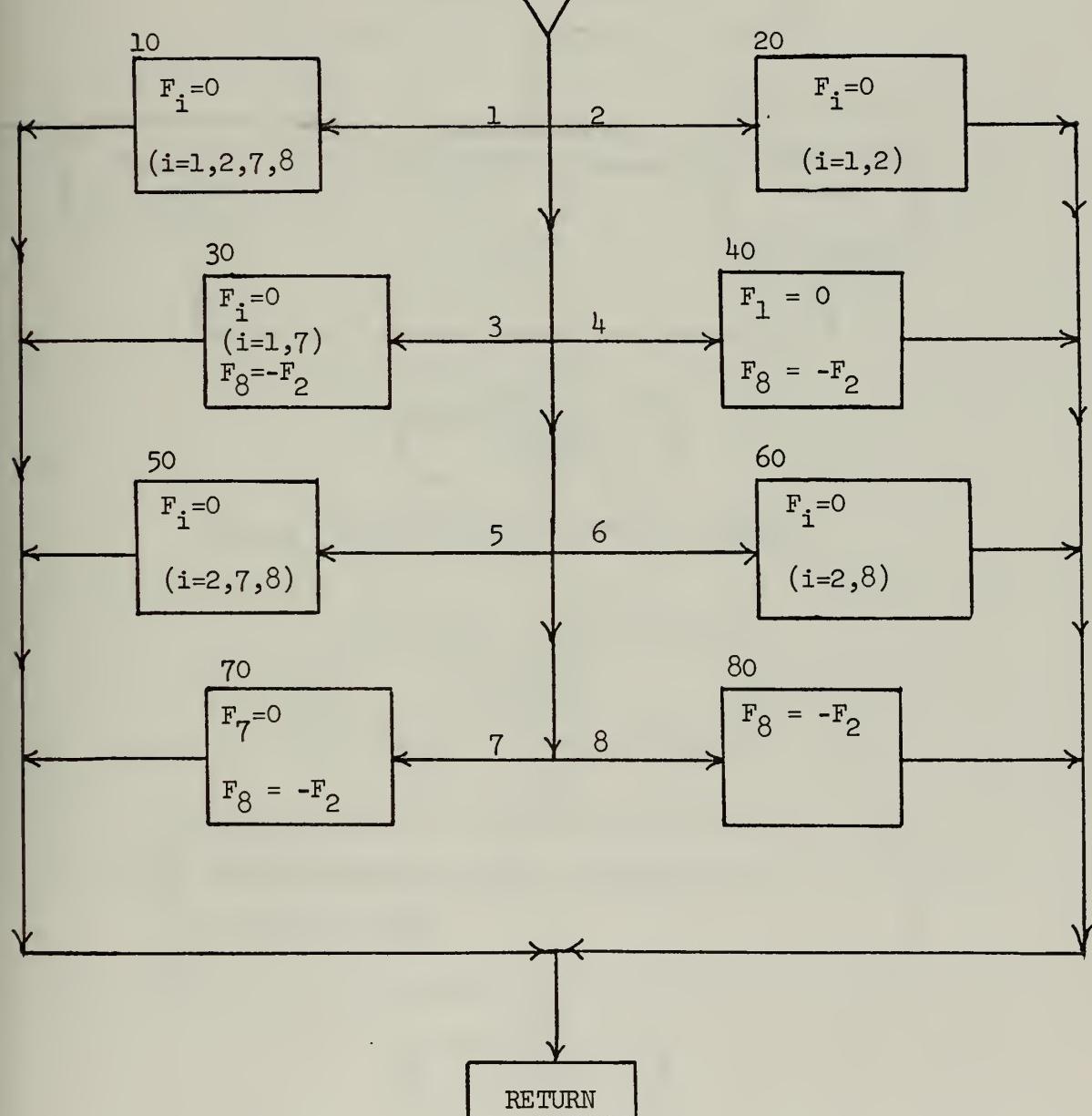


Figure 4.21 Subroutine FIXFOR Flow Diagram

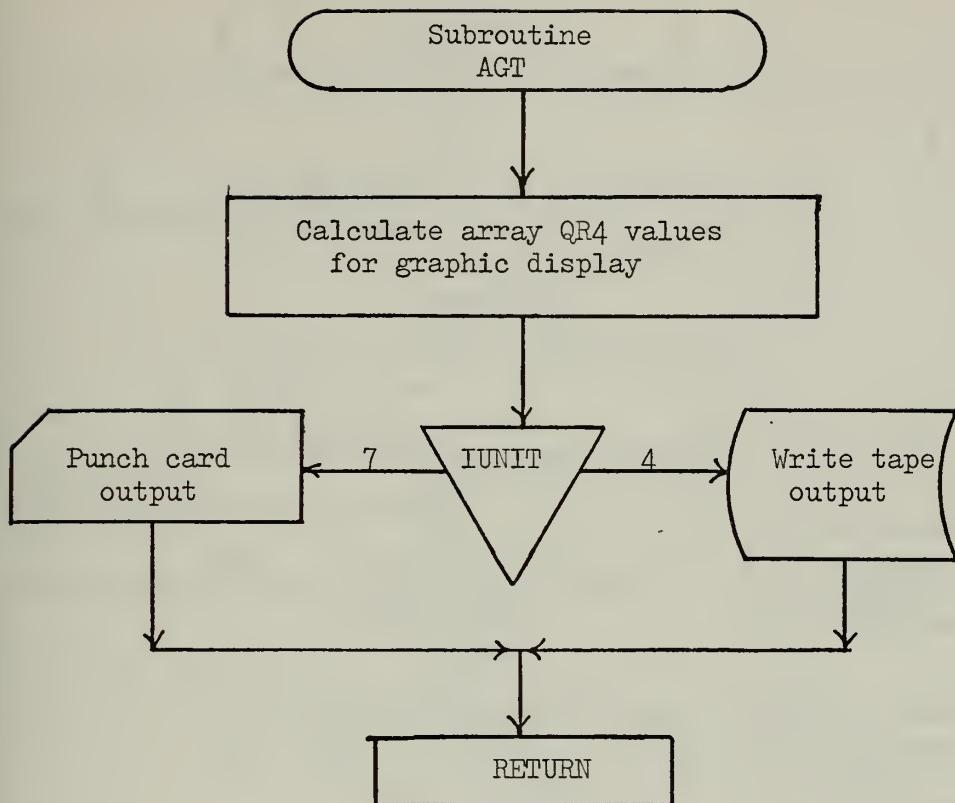


Figure 4.22 Subroutine AGT Flow Diagram

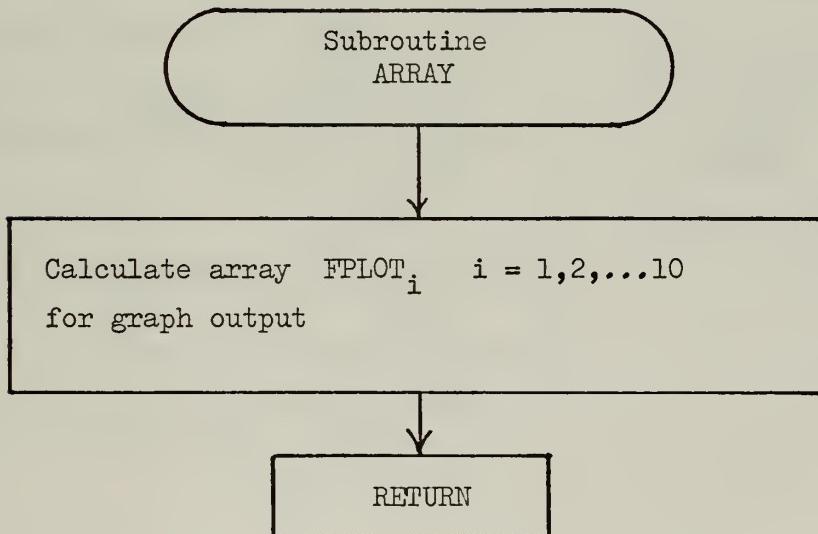


Figure 4.23 Subroutine ARRAY Flow Diagram

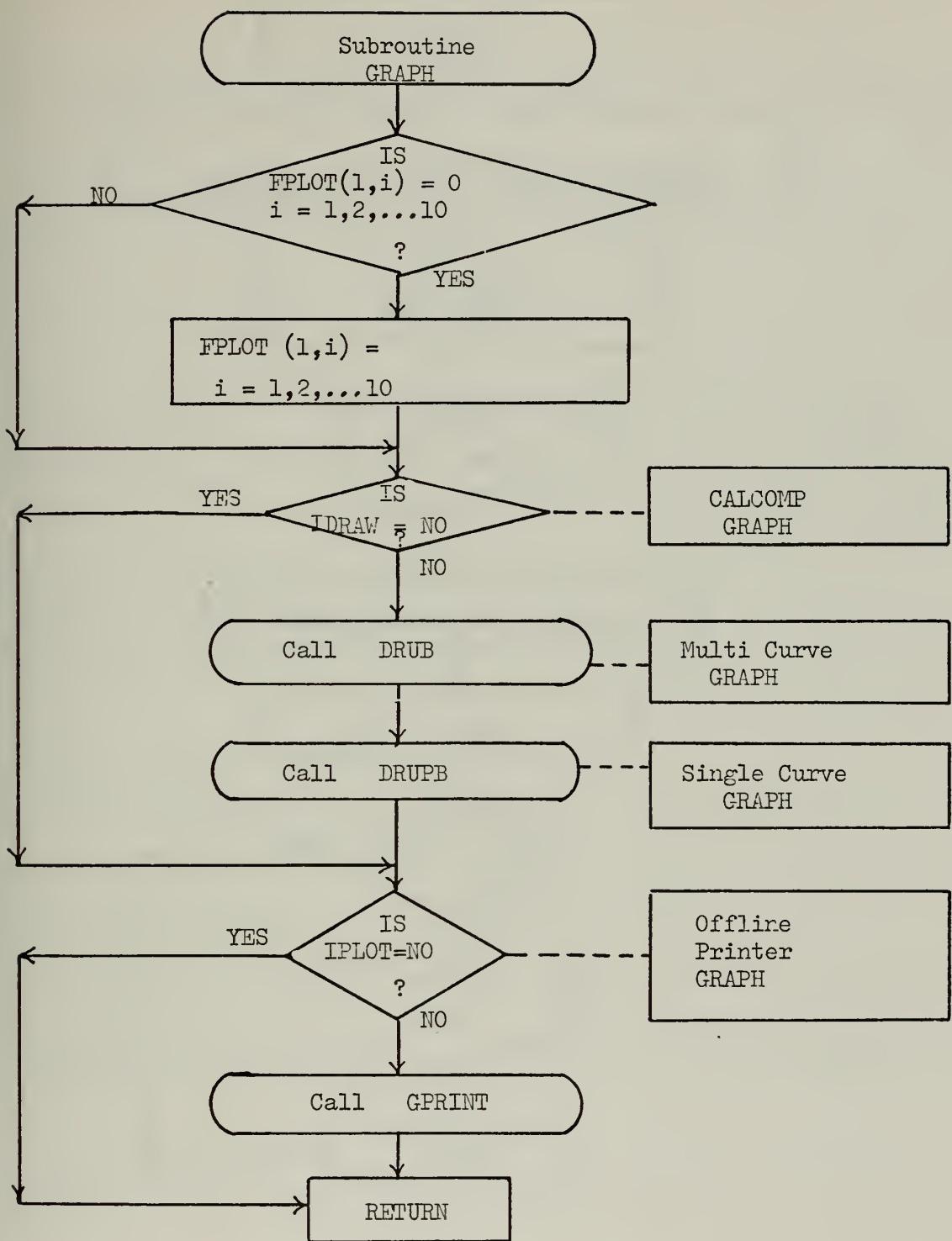


Figure 4.24 Subroutine GRAPH Flow Diagram

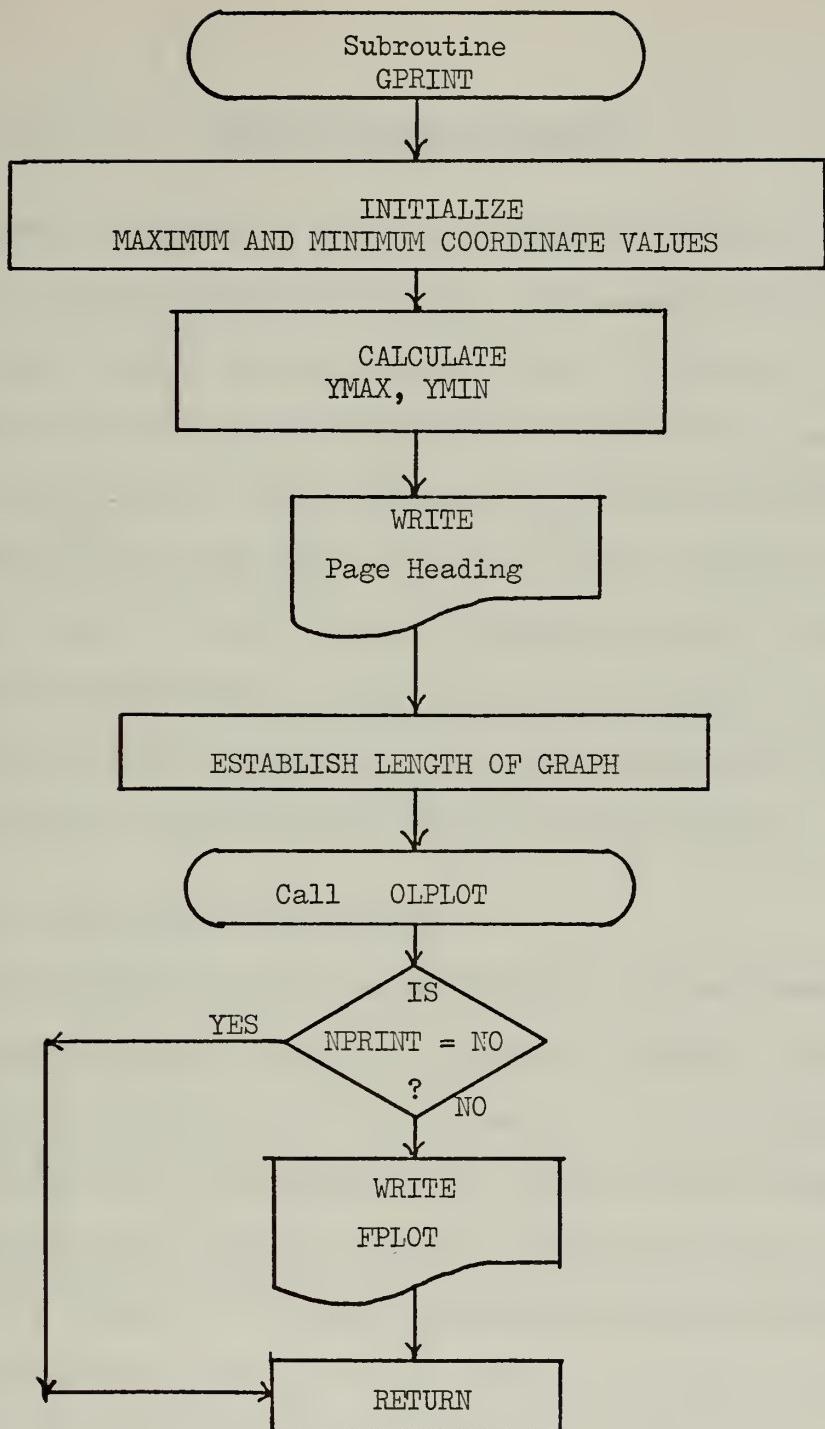


Figure 4.25 Subroutine GPRINT Flow Diagram

V. SDS-9300 COMPUTER PROGRAM

This computer program is coded in SDS FORTRAN IV, (Ref. 5-7), to run on a Scientific Data System Model 9300 computer interfaced with an ADAGE Graphics Display Terminal, Model 10 (AGT-10). This program provides the user the capability to observe the dynamic response of 360 degrees of cam rotation of the valve train assembly. The displacement data generated by the previously described program is the major input. Finite element techniques are used to reformulate the input displacement data to provide for the pictoral responses of bending and axial deformations. Due to the limited core storage of this computer, overlaying techniques have been employed.

A. METHOD OF ELASTICITY CALCULATIONS

As shown in Figure 2.4, the system consists of five elements. Three of these elements, (1,4, and 5) are axial members. The remaining two, (2 and 3) are bending members. Thus the elastic mode shapes for each of the elements as a function of the distance along the member will yield the required deformations. R. C. Winfrey, reference (1) Chapter V, has done considerable work in the formulation of the required mode shapes by finite element techniques. The mode shapes for the axial elements as a function of x are:

$$\Phi_1(x) = \frac{x}{L} \quad (5.1)$$

$$\Phi_2(x) = \frac{x}{L} - 1 \quad (5.2)$$

where L is the longitudinal dimension of the bar.

The mode shapes for the bending elements as a function of x are:

$$\Phi_1(x) = \frac{1}{L^3} (L^3 - 3x^2 L + 2x^3) \quad (5.3)$$

$$\Phi_2(x) = \frac{1}{L^3} (3x^2 L - 2x^3) \quad (5.4)$$

$$\Phi_3(x) = \frac{1}{L^2} (xL^2 - 2x^2 L + x^3) \quad (5.4)$$

$$\Phi_4(x) = \frac{1}{L^2} (x^2 L - x^3) \quad (5.6)$$

where L is the longitudinal dimension of the bar.

The system displacements are calculated by the IBM-360 program and sent as data to the SDS-9300. These displacements and the above mode shapes are superimposed to yield the required rigid body motion and elastic deformations.

B. PROGRAM STRUCTURE

1. General Description

The main program controls the program execution. The calling order is presented in Figure 5.1. The subroutines indicated by an asterick are supplied by the computer laboratory, and are used to manipulate and initialize data sent to the ADAGE Graphics Terminal. The SDS computer total core available for program execution, exclusive of the control and real time monitoring functions, is approximately 18 K bytes. Since the total program length greatly exceeds the core available, a program overlay technique consisting of storing only active subroutines during execution is employed. This achieves maximum use of the main core storage and allows the program to execute.

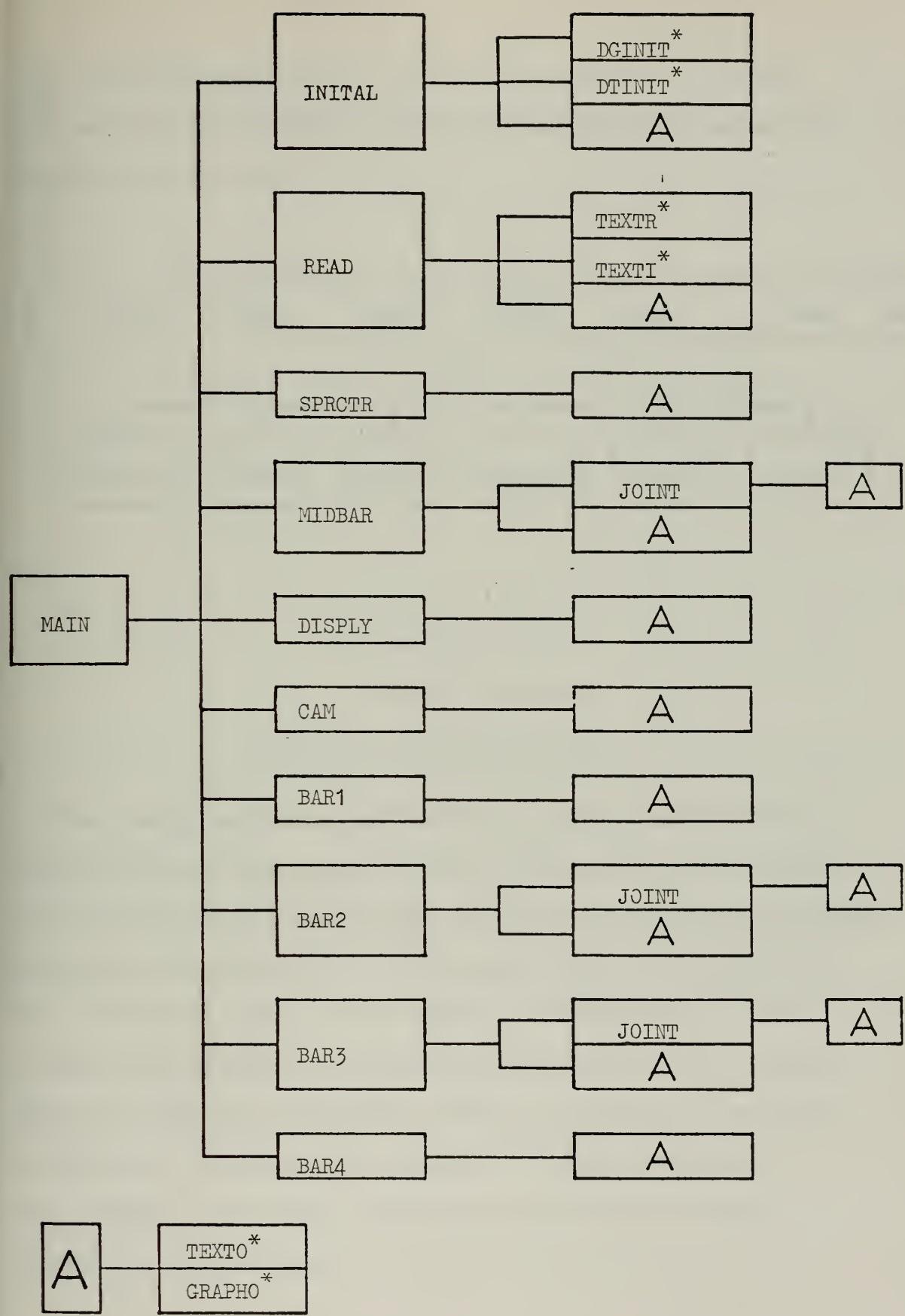


Figure 5.1 SDS-9300 Computer Program Structure

The overlay tree structure is depicted in Figure 5.2. Although necessary for this program, overlaying significantly increases the program execution time.

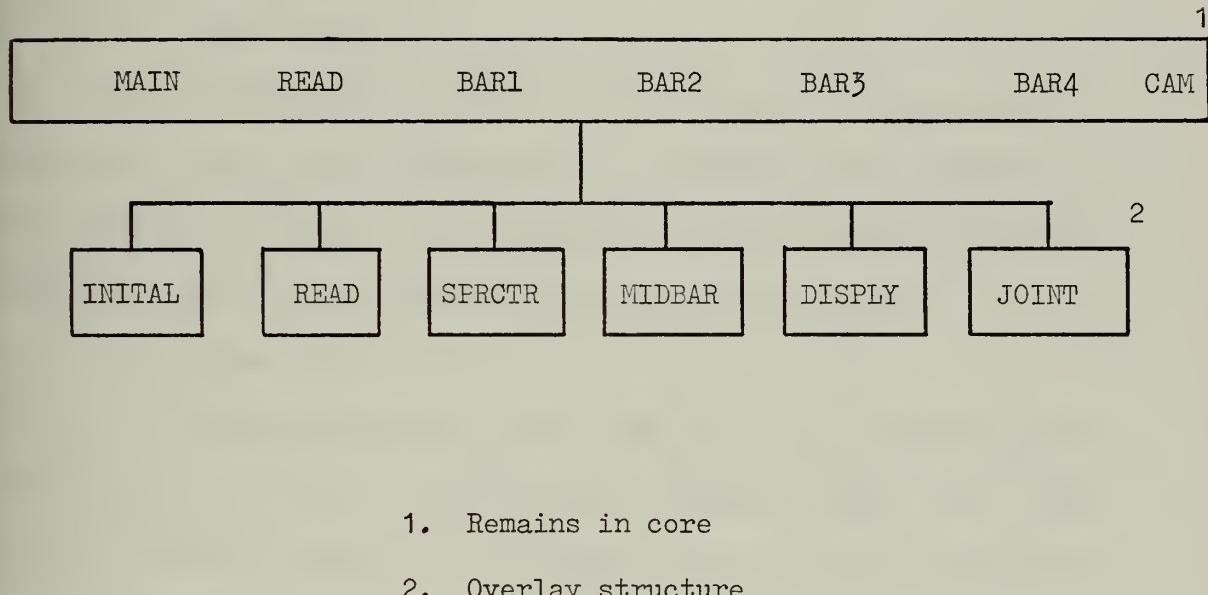


Figure 5.2 Overlay Structure

The bowing and thinning of the axial elements, as seen on the graphics display photographs, have been added as an aid in observing the deformations of these members. The mode shapes given by equations 5.1 and 5.2 are only valid for the elastic line of a one dimensional bar. However the small uniform extension and compression of the element were impossible to satisfactorily reproduce on the display. Thus the bowing and thinning were added as a function of the axial deformation. The thinner the bar, the more axial compression it has undergone. Conversely, the thicker the bar the more axial extension it has undergone.

2. Sense Lights

The SDS computer being a hands-on system, has several physical sense lights. When lit, logic within the program may alter computer instructions as desired.

a. Sense Light 1.

Activation of sense light one branches the execution to the head of the program. Execution then continues and variables are reinitialized as coded in the program. New data points may also be read in, displaying a different set of displacement data.

b. Sense Light 2.

During program execution this sense light indicates that the display will utilize ADAGE Graphics Terminal, number two. When not lit, the SDS computer is interfaced with the AGT unit number one.

c. Sense Light 4.

This sense light branches the program to a convenient point to reinitialize the originally coded program parameters. This is accomplished by activation of the SDS control console teletype from which the user may input any program variables that appear in the namelist statement.

d. Sense Light 6.

Activation of sense light six calls the subroutine TEXTR for each cam rotation increment. This enables the motion of the display to be halted for further study at any point of interest in cam rotation. Depression of the return key on the AGT terminal will advance the display to the next display of cam rotation.

C. MAIN PROGRAM

The main program initializes parameters, establishes the data arrays and contains the calculation loop. Prior to the execution of the program the desired sense lights should be activated. The program is halted almost immediately after execution by an INPUT statement. This statement allows the user to have positive control over assignment of the AGT unit. The designation of the wrong unit at this point will cause a core image failure. The program then must be reloaded. Therefore at this point the 9300 teletype control console should be checked to insure the correct unit is listed in the message to the user. To proceed with the program execution a "*" must be typed on the control console followed by the carriage return key.

Subroutines are called to display the fixed elements of the model. The two arrays that reproduce the circular cam are pre-calculated. Next the frequency ratio is calculated and displayed. The cycle loop is then entered where subroutines are called to display the movable elements. This loop is incremented by a specified amount of cam rotation until 360 degrees of rotation has been reached. At this point the cycle is repeated. For each cam increment, the sense lights are tested.

D. SUBROUTINES

The function of each of the subroutines and a description of their execution are included. Each subroutine is listed in its entirety in the SDS-9300 Computer Program Listing.

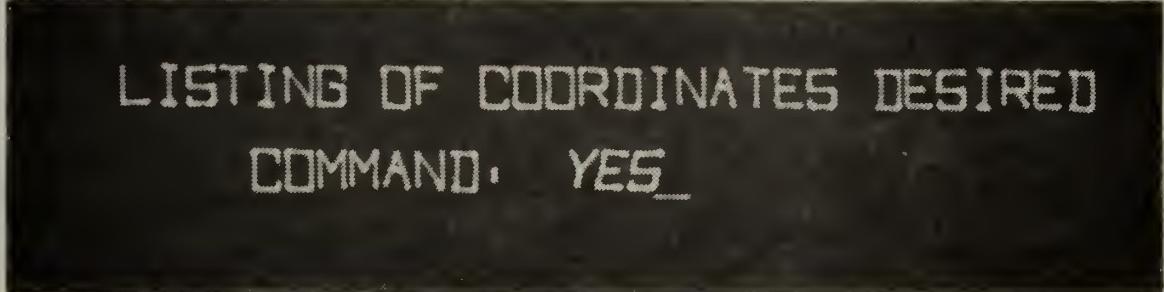
1. Subroutine INITIAL

This subroutine initializes the text and graphics directory arrays, DTINIT and DGINIT, respectively. The text response, TEXTR is initialized to block number one at display location 39, 46. The title display is encoded and presented on the AGT. The graphic data blocks are initialized to zero, assigned the appropriate first word header (IHEAD), and then assigned sequence numbers in DGINIT for later use.

2. Subroutine READ

This subroutine reads the title and label for the program and prints them on the offline printer. The program parameters are read, (XZERO, AMP, RPM, WN, ISEAT, and PIN28) and also printed for reference. The user is then questioned from the AGT display,

Photo 5.1.



LISTING OF COORDINATES DESIRED
COMMAND: YES

Photo 5.1 AGT QUESTION

An affirmative response will list the IBM-360 displacement data on the offline printer. The response is entered via the graphics terminal.

The displacement data is next read from magnetic tape or punched cards, depending on the control specification at the time the program was loaded. The array parameters read are, Q1 through Q9, OMEGA, and CONFIG. The Q's are as defined in the IBM program but

modified by the amount XZERO. OMEGA is the amount of revolution of the cam in radians. CONFIG is the corresponding configuration number. Next the displacements are printed, if desired. Finally the title screen is nulled and control is returned to the main program.

3. Subroutine SPRCTR

This subroutine is only called once for a given problem. The display for the rigid portion of the spring is calculated and displayed. The valve seat position and geometry is calculated. The parameter ISEAT is then checked to determine whether the problem considers contact at the valve seat. For those problems with no contact, the valve seat location is adjusted to better picture the position. Finally the valve seat is displayed and control returns to the main program.

4. Subroutine MIDBAR

This subroutine is called once. Subroutine MIDBAR calculates and displays the fixed pivot point.

5. Subroutine DISPLAY

This subroutine is called once during each cam increment. The function of this routine is to display the text for the system configurations.

6. Subroutine CAM

Subroutine CAM is called once during each cam increment. This routine calculates the cam position, given the cam rotation (OMEGA) and displacement of the top of the cam (q_9).

7. Subroutine BAR1

This subroutine is called once during each cam increment.

Coordinates q_1 and q_8 are used to calculate the position of the pushrod. Equations 5.1 and 5.2 calculate the deformations. The pushrod is then displayed on the AGT.

8. Subroutine BAR2

This subroutine is called once during each cam increment.

Coordinates q_2 , q_3 , and q_6 , and equations 5.3, 5.4, and 5.5 are used in the calculations of position, geometry, and deformations of the left rocker arm. The parameter PIN28 is checked to determine whether the pin should be displayed at clearance two. The left rocker arm and pin, if required, are displayed on the AGT.

9. Subroutine BAR3

This subroutine is called once during each cam increment.

Coordinates q_4 , q_5 , and q_6 , and equations 5.3, 5.4, and 5.5 are used in the calculation of position, geometry, and deformations of the right rocker arm. Calculation of the position and image of the rocker arm- valve stem- valve spring joint pin is accomplished by subroutine JOINT. The right rocker arm and pin are then displayed on the AGT.

10. Subroutine BAR4

This subroutine is called once during each cam increment.

Coordinates q_4 and q_7 are used to calculate the position of the valve stem. Equations 5.1 and 5.2 calculate the deformations. The valve- valve stem is then displayed on the AGT.

11. Subroutine JOINT

This subroutine is called once for the right rocker arm pin and, if required, once for the left rocker arm pin, during each cam increment. The pins' position and image are calculated and displayed on the AGT.

VI. NUMERICAL RESULTS

The precalculated system matrices for the physical properties assumed in chapter 2, are presented in this chapter. Numerical examples have been chosen for their interest and for comparison with one another. The relationship between examples is indicated in Figure 6.26. Example results are presented both as CALCOMP graphs and/or AGT display photos. Photo 6.1 is the title page from the graphics terminal display.

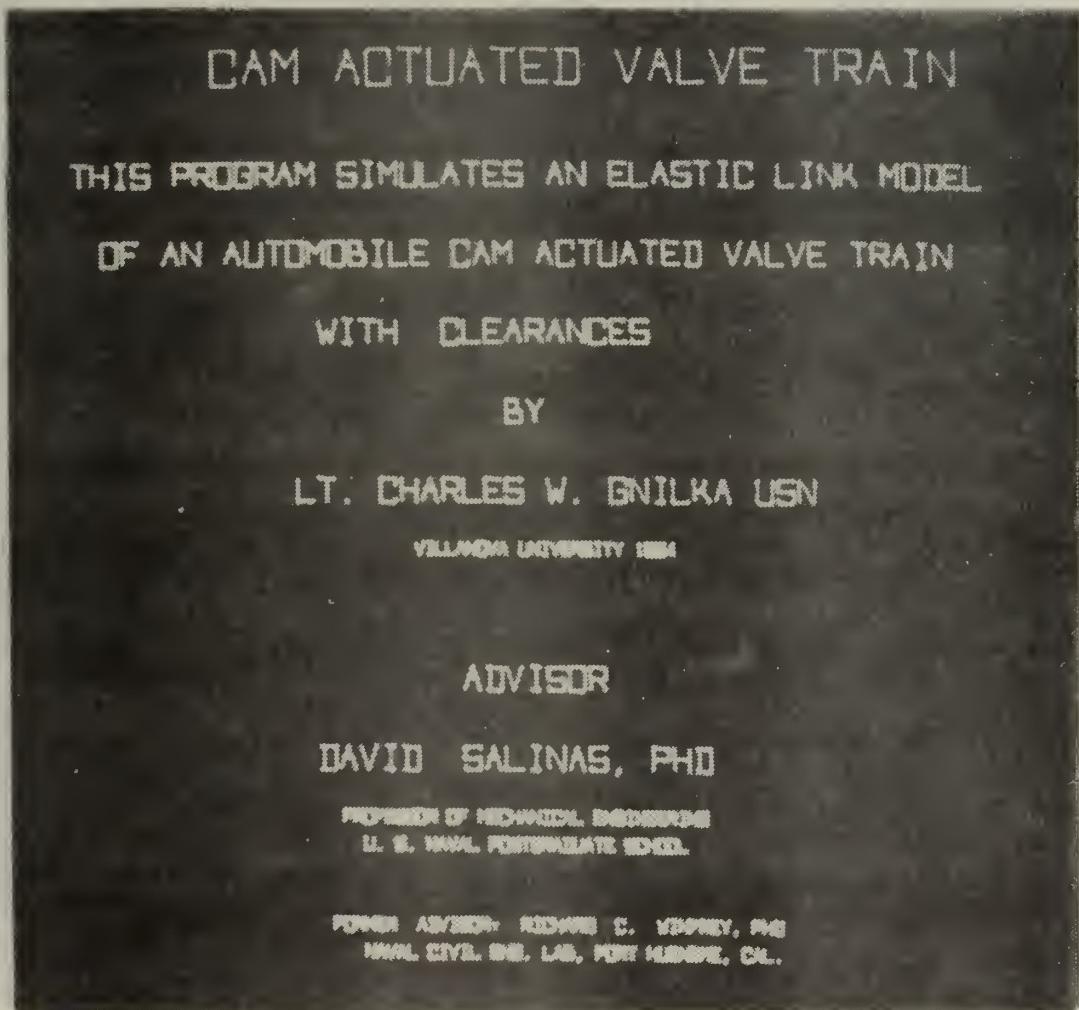


Photo 6.1 AGT Title Page

A. MODE SHAPES AND NATURAL FREQUENCIES

The mode shapes and natural frequencies presented in Figure 6.1 result from the eigenvalue problem for the full eight coordinate system. Figure 6.2 shows the values resulting from the seven coordinate reduced system, for contact at the rocker.

NATURAL FREQUENCIES ω_n (EIGENVALUES) $\times 10^3$							
0.0	63.6	638.7	0.886	263.8	1371.1	149.8	77.86
MODE SHAPES (EIGENVECTORS) IN COLUMNS [Φ]							
46.3	0.0	0.0	0.0	0.0	0.0	0.0	-80.2
0.0	115.0	-109.5	-37.1	-57.2	-172.7	33.3	0.0
0.0	-131.3	581.0	37.1	149.1	1538.0	-56.0	0.0
0.0	21.5	-34.2	55.6	60.9	18.7	42.5	0.0
0.0	58.7	-499.4	37.0	347.7	376.9	-29.8	0.0
0.0	-70.9	-340.0	37.1	-146.7	381.4	24.6	0.0
0.0	-35.0	-18.9	-55.6	52.8	9.6	120.0	0.0
46.3	0.0	0.0	0.0	0.0	0.0	0.0	80.2

Figure 6.1 FULL SYSTEM MODE SHAPES AND NATURAL FREQ.

NATURAL FREQUENCIES ω_n (EIGENVALUES) $\times 10^3$							
0.691	73.6	521.9	25.9	247.1	1070.2	146.7	
MODE SHAPES (EIGENVECTORS) IN COLUMNS [Φ]							
28.96	-69.26	15.64	45.72	11.22	12.01	-8.39	
28.95	64.03	-30.77	20.91	-20.84	-23.92	13.67	
-28.95	-70.49	503.13	-30.45	136.63	799.47	-42.66	
-43.37	9.09	-34.11	39.89	61.27	25.02	38.35	
-28.90	32.45	-427.34	39.72	313.46	471.17	-45.62	
-28.93	-42.34	-215.68	-1.53	-164.75	436.79	44.23	
43.37	-18.08	-19.75	-42.97	57.06	12.96	118.61	

Figure 6.2 REDUCED SYSTEM MODE SHAPES AND NATURAL FREQ.

B. PRE-CALCULATED SYSTEM MATRICES

The remaining system matrices are presented and grouped according to their layer (or configuration) number. The mass, stiffness, and damping matrices for layer one are unaltered and directly correspond to the full eight coordinate system eigenvalue problem. The mass, stiffness, and damping matrices in layer three are unaltered and directly correspond to the reduced seven coordinate system eigenvalue problem.

The pre-calculated matrices COLM, COLC, and COLK located in layers one and three are associated with the calculations for configurations six and eight respectively. Thus, these matrices are not listed in their correct program layer, but with the correct configuration with which they are used.

The order of presentation within each layer is: (1) Mass, (2) Damping, (3) Stiffness, (4) COLM, (5) COLC, (6) COLK, (7) BMIV, (8) DA, and (9) DV.

CONFIGURATION AND MATRIX LAYER = 1
 MATRICES ARE 8X8

--- SYSM ---							
1.55D-04	0.0	0.0	0.0	0.0	0.0	0.0	7.77D-05
0.0	7.70D-05	1.09D-05	0.0	0.0	-6.42D-06	0.0	0.0
--- DAMP ---							
6.05D-01	0.0	0.0	0.0	0.0	0.0	0.0	-6.05D-01
0.0	1.32D-01	2.93D-01	5.54D-01	-1.69D-01	2.49D-01	-7.00D-02	0.0
0.0	2.93D-01	1.42D-01	1.00D-01	-3.70D-02	2.01D-02	-1.01D-02	0.0
0.0	5.54D-01	1.90D-01	1.51D-01	-3.36D-01	-5.39D-02	9.12D-01	0.0
--- STF ---							
2.36D 05	0.0	0.0	0.0	0.0	0.0	1.15D 06	0.0
0.0	2.29D 06	1.15D 06	0.0	0.0	1.82D 05	0.0	-2.36D 05
0.0	1.15D 06	7.63D 05	0.0	0.0	-5.09D 05	0.0	0.0
0.0	0.0	0.0	1.39D 06	-5.09D 05	2.54D 05	7.07D 05	0.0
0.0	0.0	0.0	-5.09D 05	2.54D 05	1.27D 06	0.0	0.0
-2.36D 05	0.0	0.0	0.0	0.0	0.0	7.07D 05	0.0
-2.36D 05	0.0	0.0	0.0	0.0	0.0	2.36D 05	0.0

Figure 6.3 CONFIGURATION I - Mass, Damping, and Stiffness

CONFIGURATION AND MATRIX LAYER = 1
 MATRICES ARE 8X8

- (BMIV) -

8.52D 03	0.0 0	0.0 0	-2.66D 05	-4.90D 02	0.0 0	-1.14D 04	0.0 0	-1.31D 04	-1.0 0	-2.34D 02	-4.23D 03
0.0 0	4.99D 04	-2.66D 05	1.97D 06	7.95D 03	1.02D 04	4.24D 04	1.89D 05	2.18D 05	3.94D 03	0.0 0	0.0 0
0.0 0	-2.66D 05	1.97D 06	7.95D 03	1.02D 04	4.24D 04	4.49D 04	8.16D 04	8.16D 03	4.96D 03	0.0 0	0.0 0
0.0 0	-4.90D 02	-4.90D 05	1.89D 05	8.16D 03	1.96D 03	1.97D 05	1.97D 05	2.30D 05	2.09D 04	0.0 0	0.0 0
0.0 0	-1.14D 04	2.18D 05	4.24D 04	4.49D 04	4.96D 04	4.04D 03	4.04D 03	2.15D 04	2.04D 03	0.0 0	0.0 0
0.0 0	-1.31D 04	3.94D 03	4.96D 03	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	8.52D 03	8.52D 03
-4.23D 03	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0

- (DA) -

7.72D 03	0.0 0	0.0 0	-2.0 31D 04	-4.0 56D 03	0.0 0	-6.0 00D 02	0.0 0	1.0 73D 03	-9.0 68D 02	-7.0 72D 03	
0.0 0	-1.0 35D 04	2.0 01D 05	-8.0 87D 03	4.0 95D 03	6.0 75D 03	6.0 77D 03	6.0 55D 02	6.0 25D 04	1.0 29D 03	0.0 0	0.0 0
0.0 0	-2.0 53D 05	5.0 51D 03	5.0 51D 02	5.0 75D 03	6.0 27D 04	8.0 42D 04	6.0 15D 04	9.0 68D 03	9.0 68D 03	0.0 0	0.0 0
0.0 0	1.0 81D 03	3.0 66D 04	1.0 52D 04	1.0 52D 04	-6.0 27D 04	2.0 83D 04	9.0 52D 04	2.0 01D 04	2.0 01D 04	0.0 0	0.0 0
0.0 0	3.0 66D 04	7.0 47D 04	2.0 53D 04	2.0 53D 04	-4.0 78D 04	2.0 01D 04	1.0 26D 03	1.0 42D 03	1.0 42D 03	0.0 0	0.0 0
0.0 0	7.0 47D 04	-4.0 46D 02	7.0 92D 01	7.0 92D 01	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0
-7.0 72D 03	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0

- (DV) -

3.00D 09	0.0 0	0.0 0	-1.0 51D 11	-1.0 16D 10	0.0 0	-8.0 87D 09	0.0 0	-6.0 36D 10	0.0 0	-5.0 12D 08	-3.00D 09
0.0 0	-2.0 05D 11	1.0 90D 12	-1.0 28D 12	-1.0 93D 11	1.0 47D 11	1.0 47D 11	1.0 85D 10	1.0 70D 11	8.0 40D 09	0.0 0	0.0 0
0.0 0	1.0 90D 12	8.0 62D 10	8.0 62D 09	-8.0 15D 09	1.0 53D 11	1.0 53D 11	1.0 00D 11	1.0 00D 11	1.0 48D 10	0.0 0	0.0 0
0.0 0	4.0 15D 11	2.0 05D 11	-2.0 51D 11	-2.0 51D 11	1.0 54D 11	1.0 54D 11	4.0 07D 11	4.0 07D 11	8.0 62D 09	0.0 0	0.0 0
0.0 0	4.0 83D 11	2.0 39D 11	-2.0 03D 11	-2.0 03D 11	9.0 38D 09	9.0 13D 09	9.0 16D 09	1.0 87D 10	1.0 87D 10	0.0 0	0.0 0
0.0 0	8.0 59D 09	4.0 28D 09	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	0.0 0	3.00D 09	3.00D 09

Figure 6.4 CONFIGURATION I - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 2
 MATRICES ARE 7X7

---- SYSM ----						
1.55D-04	0.0	0.0	0.0	0.0	0.0	7.77D-05
0.0	7.70D-05	1.09D-05	1.09D-05	0.0	0.0	-6.42D-06
---- DAMP ----						
6.05D-01	0.0	0	0	0	0	-6.05D-01
0.0	1.32D-00	2.93D-01	5.54D-01	-1.69D-01	2.49D-01	0.0
0.0	2.93D-01	1.42D-01	1.00D-01	-3.70D-02	2.01D-02	0.0
0.0	5.54D-01	1.00D-01	1.51D-00	-3.36D-01	-5.39D-02	0.0
0.0	-1.69D-01	-3.70D-02	-3.36D-01	2.71D-01	-6.93D-02	0.0
0.0	2.49D-01	2.01D-02	-5.39D-02	-6.93D-02	4.69D-01	0.0
-6.05D-01	0.0	0.0	0.0	0.0	0.0	6.05D-01
---- STF ----						
2.36D 05	0.0	0.0	0.0	0.0	0.0	-2.36D 05
0.0	2.29D 06	1.15D 06	1.15D 06	0.0	1.15D 06	0.0
0.0	1.15D 06	7.63D 05	0.0	0.0	3.82D 05	0.0
0.0	0.0	0.0	1.39D 06	-5.09D 05	-5.09D 05	0.0
0.0	0.0	0.0	-5.09D 05	5.09D 05	2.54D 05	0.0
-2.36D 05	0.0	0.0	0.0	0.0	1.27D 06	0.0
					0.0	2.36D 05

Figure 6.5 CONFIGURATION II - Mass, Damping, and Stiffness

(This page intentionally blank)

CONFIGURATION AND MATRIX LAYER = 2
MATRICES ARE 7x1

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
0.0	0.0	0.0
0.0	-7.00D-02	0.0
0.0	-1.01D-02	0.0
-2.59D-05	9.12D-01	7.07D 05
0.0	-1.09D-01	0.0
0.0	5.66D-02	0.0
0.0	0.0	0.0

Figure 6.6 CONFIGURATION II - Precalculated Matrices; COLM, COLC, and COLK

CONFIGURATION AND MATRIX LAYER = 2

MATRICES ARE 7X7

→ (BMIV) -

8.52D 03	0.0	0.0	-2.66D 05	-4.36D 02	0.0	-1.11D 04	0.0	-1.31D 04	0.0	-4.23D 03
0.0	4.99D 04	-2.66D 05	1.97D 06	7.04D 03	-1.03D 03	1.85D 05	-1.17D 05	2.17D 05	0.0	0.0
0.0	-2.66D 05	0.2	7.04D 03	9.03D 03	3.76D 04	7.23D 05	7.23D 05	0.0	0.0	0.0
0.0	-4.36D 02	0.4	1.85D 05	3.76D 04	4.20D 05	1.93D 05	2.90D 05	0.0	0.0	0.0
0.0	1.11D 04	2.17D 05	7.23D 03	0.0	1.93D 05	2.90D 05	0.0	0.0	0.0	0.0
0.0	-1.31D 04	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-4.23D 03	-1.31D 04	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

- (DA) -

7.72D 03	0.0	0.0	-2.31D 04	4.78D 03	0.0	-5.86D 02	0.0	1.74D 03	0.0	-7.72D 03
0.0	-1.35D 04	-2.53D 05	2.01D 05	-1.26D 04	4.71D 03	6.22D 04	6.22D 04	0.0	0.0	0.0
0.0	2.53D 05	0.3	5.32D 02	1.10D 03	6.48D 03	3.28D 02	3.28D 02	0.0	0.0	0.0
0.0	1.91D 03	0.4	1.51D 04	-8.22D 04	8.30D 04	6.01D 04	6.01D 04	0.0	0.0	0.0
0.0	3.70D 04	0.4	2.53D 04	-5.16D 04	2.81D 04	9.49D 04	9.49D 04	0.0	0.0	0.0
0.0	7.48D 04	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-7.72D 03	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

- (DV) -

3.00D 09	0.0	0.0	-1.51D 11	1.17D 10	0.0	-8.77D 09	0.0	-6.35D 10	0.0	-3.00D 09
0.0	-2.05D 11	-1.90D 12	-1.28D 12	-1.95D 11	1.46D 11	7.68D 11	7.68D 11	0.0	0.0	0.0
0.0	1.53D 10	1.0	7.64D 11	-1.03D 10	1.64D 10	1.64D 10	1.64D 10	0.0	0.0	0.0
0.0	4.07D 11	1.1	2.02D 11	-2.60D 11	2.44D 11	3.91D 11	3.91D 11	0.0	0.0	0.0
0.0	4.82D 11	0.0	2.39D 11	-2.05D 11	1.53D 11	4.05D 11	4.05D 11	0.0	0.0	0.0
-3.00D 09	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 6.7 CONFIGURATION III - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 3
MATRICES ARE 7X7

--- SYSM ---						
--- DAMP ---						
--- STF ---						
1.55D-04	7.77D-05	0.0	0.0	0.0	0.0	0.0
7.77D-05	2.32D-04	1.97D-05	0.0	0.0	-6.42D-06	0.0
0.0	1.09D-05	1.97D-06	0.0	-2.44D-05	-1.44D-05	-2.59D-05
0.0	0.0	0.0	-2.44D-05	6.66D-06	-5.00D-06	0.0
0.0	0.0	-1.48D-06	-1.44D-05	-5.00D-06	8.64D-06	0.0
0.0	-6.42D-06	-1.48D-06	-2.59D-05	0.0	0.0	5.18D-05
0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.48D 00	1.05D 00	-7.37D-03	1.72D 00	-1.60D-01	9.80D-02	-6.62D-01
1.05D 00	3.49D 00	3.58D-01	2.09D 00	-3.08D-01	3.16D-01	-6.69D-01
-7.37D-03	3.58D-01	2.03D-01	1.02D-01	-3.48D-02	2.59D-03	-1.65D-02
-1.72D 00	2.09D 00	1.02D-01	3.10D 00	-4.83D-01	3.25D-02	-2.99D-01
-1.60D-01	-3.08D-01	-3.48D-02	-4.83D-01	2.886D-01	-8.02D-02	-5.40D-02
-9.80D-02	-3.16D-01	-2.59D-03	3.25D-02	-8.02D-02	4.91D-01	2.90D-02
-6.62D-01	-6.69D-01	-1.65D-02	2.99D-01	-5.40D-02	2.90D-02	1.17D 00
2.36D 05	-2.36D 05	0.0	0.0	0.0	0.0	0.0
-2.36D 05	2.53D 06	1.15D 06	0.0	0.0	1.15D 06	0.0
0.0	1.15D 06	7.63D 05	0.0	0.0	3.82D 05	0.0
0.0	0.0	0.0	1.39D 06	-5.09D 05	-5.09D 05	7.07D 05
0.0	0.0	0.0	-5.09D 05	5.09D 05	2.54D 05	0.0
0.0	1.15D 06	3.82D 05	7.07D 05	2.54D 05	1.27D 06	0.0
0.0	0.0	0.0	0.0	0.0	0.0	7.07D 05

Figure 6.8 CONFIGURATION III - Mass, Damping, and Stiffness

MATRICES ARE 7X7

- (BMIV) -						
8•18D	03	-3•62D	03	1•90D	04	-1•34D
-3•62D	03	-7•24D	04	-7•89D	01	-1•58D
-1•90D	04	-3•81D	04	5•68D	05	1•35D
1•34D	01	-7•89D	01	5•68D	03	4•23D
7•67D	02	-1•58D	03	4•23D	04	4•23D
9•06D	02	-1•82D	03	1•56D	05	8•03D
3•82D	01	-2•14D	01	1•56D	03	1•94D
				2•79D	03	2•97D
				03	04	05
- (DA) -						
1•63D	04	2•88D	03	2•48D	03	8•18D
-1•09D	03	7•55D	03	-5•07D	03	5•46D
-3•14D	03	1•73D	05	-1•35D	05	-1•18D
8•08D	03	9•12D	03	6•32D	02	1•32D
7•20D	03	4•42D	04	1•61D	04	-15•59D
1•65D	03	7•66D	04	2•57D	04	-4•63D
-8•56D	03	-8•19D	03	-4•62D	-01	1•22D
				04	04	04
- (DV) -						
2•78D	09	-1•18D	10	1•07D	10	-8•06D
-2•56D	09	-2•66D	10	-2•15D	10	-1•61D
-1•35D	10	-9•37D	11	5•88D	11	-1•38D
2•17D	07	1•55D	10	7•31D	09	-8•05D
5•54D	08	3•72D	11	1•75D	11	-2•48D
6•43D	08	4•34D	11	2•04D	11	-2•00D
1•41D	07	7•69D	09	3•63D	09	9•44D
				09	09	09
					09	09
					09	09
					09	09

Figure 6.9 CONFIGURATION III - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 4

MATRICES ARE 6X6

---- SYSM ----

1•55D-04	7•77D-05	0•0	0•0	0•0	0•0
7•77D-05	2•32D-04	1•09D-05	0•0	0•0	-6•42D-06
0•0	1•09D-05	1•97D-06	0•0	0•0	-1•48D-06
0•0	0•0	0•0	2•01D-04	-2•44D-05	1•44D-05
0•0	0•0	-2•44D-05	6•66D-06	-5•00D-06	8•64D-06
0•0	-6•42D-06	-1•48D-06	1•44D-05	-5•00D-06	

---- DAMP ----

2•48D 00	1•05D 00	-7•37D-03	1•72D 00	-1•60D-01	9•80D-02
1•05D 00	3•49D 00	3•58D-01	2•09D 00	-3•08D-01	3•16D-01
-7•37D-03	3•58D-01	2•03D-01	1•02D-01	-3•48D-02	2•59D-03
-1•72D 00	2•09D 00	1•02D-01	3•10D 00	-4•83D-01	3•25D-02
-1•60D-01	-3•08D-01	-3•48D-02	-4•83D-01	2•86D-01	-8•02D-02
-9•80D-02	3•16D-01	2•59D-03	3•25D-02	-8•02D-02	4•91D-01

---- STF ----

2•36D 05	-2•36D 05	0•0	0•0	0•0	0•0
-2•36D 05	2•53D 06	1•15D 06	0•0	0•0	1•15D 06
0•0	1•15D 06	7•63D 05	0•0	0•0	3•82D 05
0•0	0•0	0•0	1•39D 06	-5•09D 05	-5•09D 05
0•0	0•0	0•0	-5•09D 05	5•09D 05	2•54D 05
0•0	1•15D 06	3•82D 05	-5•09D 05	2•54D 05	1•27D 06

Figure 6.10 CONFIGURATION IV - Mass, Damping, and Stiffness

CONFIGURATION AND MATRIX LAYER = 4
MATRICES ARE 6X1

COLUMN MATRICES FOR CALCULATION OF DELTA P		
COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
0.0	-6.62D-01	0.0
0.0	-6.69D-01	0.0
0.0	-1.65D-02	0.0
-2.59D-05	2.99D-01	7.07D 05
0.0	-5.40D-02	0.0
0.0	2.90D-02	0.0

Figure 6.11 CONFIGURATION IV - Precalculated Matrices; COLM, COLC, and COLK

CONFIGURATION AND MATRIX LAYER = 4

MATRICES ARE 6X6

- (BMIV) -

8.18D	03	-3.62D	03	1.90D	04	4.55D	00	7.30D	02	8.99D	02
-3.62D	03	-7.24D	03	-3.81D	04	-7.39D	01	-1.56D	03	-1.82D	03
-1.90D	04	-3.81D	04	-7.49D	05	5.03D	03	1.32D	05	1.56D	05
4.55D	00	-7.39D	01	5.03D	03	9.00D	03	3.75D	04	7.11D	03
7.30D	02	-1.56D	03	1.32D	05	3.75D	04	4.18D	05	1.90D	05
8.99D	02	-1.82D	03	1.56D	05	7.11D	03	1.90D	05	2.26D	05

- (DA) -

1.63D	04	2.90D	03	2.48D	03	8.16D	03	-7.20D	02	9.30D	01
-1.10D	03	7.55D	03	-5.07D	03	5.47D	03	-5.94D	02	1.06D	03
-4.26D	03	1.74D	05	-1.35D	05	-1.33D	04	5.52D	03	5.78D	04
1.01D	04	1.10D	04	6.32D	02	1.04D	04	5.64D	03	7.64D	02
1.55D	04	5.21D	04	1.61D	04	-6.78D	04	8.17D	04	6.07D	04
3.24D	03	7.81D	04	2.57D	04	-4.86D	04	2.76D	04	9.58D	04

- (DV) -

2.78D	09	1.18D	10	1.07D	10	-8.23D	08	5.98D	08	4.45D	09
-2.56D	09	-2.66D	10	-2.15D	10	1.62D	09	-1.22D	09	-8.93D	09
-1.35D	10	-9.36D	11	-5.88D	11	-1.40D	11	1.04D	11	4.72D	11
1.85D	07	1.37D	10	6.47D	09	-1.02D	10	1.63D	10	1.58D	10
5.40D	08	3.65D	11	1.72D	11	-2.57D	11	2.42D	11	3.77D	11
6.41D	08	4.32D	11	2.03D	11	-2.02D	11	1.50D	11	3.90D	11

Figure 6.12 CONFIGURATION IV - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 5

MATRICES ARE 7X7

--- SYSM ---						
7.70D-05	1.09D-05	0.0	0.0	-6.42D-06	0.0	0.0
1.09D-05	1.97D-06	0.0	0.0	-1.48D-06	0.0	0.0
0.0	0.0	2.01D-04	-2.44D-05	-1.44D-05	-2.59D-05	0.0
0.0	0.0	-2.44D-05	6.66D-06	-5.00D-06	0.0	0.0
-6.42D-06	-1.48D-06	-1.44D-05	-5.00D-06	8.64D-06	0.0	0.0
0.0	0.0	-2.59D-05	0.0	0.0	5.18D-05	0.0
0.0	0.0	0.0	0.0	0.0	1.55D-04	0.0
--- DAMP ---						
1.32D 00	2.93D-01	5.54D-01	-1.69D-01	2.49D-01	-7.00D-02	0.0
2.93D-01	1.42D-01	1.00D-01	-3.70D-02	2.01D-02	-1.01D-02	0.0
5.54D-01	1.00D-01	1.51D-00	-3.36D-01	-5.39D-02	-9.12D-01	0.0
-1.69D-01	-3.70D-02	-3.36D-01	-2.71D-01	-6.93D-02	-1.09D-01	0.0
-7.00D-02	-2.01D-02	-5.39D-02	-6.93D-02	4.69D-01	5.66D-02	0.0
0.0	-1.01D-02	9.12D-01	-1.09D-01	5.66D-02	9.31D-01	0.0
0.0	0.0	0.0	0.0	0.0	6.05D-01	0.0
--- STF ---						
2.29D 06	1.15D 06	0.0	0.0	1.15D 06	0.0	0.0
1.15D 06	7.63D 05	0.0	0.0	3.82D 05	0.0	0.0
0.0	0.0	1.39D 06	-5.09D 05	-5.09D 05	7.07D 05	0.0
0.0	0.0	-5.09D 05	2.54D 05	2.54D 05	0.0	0.0
1.15D 06	3.82D 05	-5.09D 05	1.27D 06	0.0	7.07D 05	0.0
0.0	0.0	7.07D 05	0.0	0.0	0.0	2.36D 05

Figure 6.13 CONFIGURATION V - Mass, Damping, and Stiffness

CONFIGURATION AND MATRIX LAYER = 5
MATRICES ARE 7X1

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
7.77D-05	-6.05D-01	-2.36D 05

Figure 6.14 CONFIGURATION V - Precalculated Matrices; COLM, COLC, and COLK

CONFIGURATION AND MATRIX LAYER = 5

MATRICES ARE 7X7

- (BMIV) -						
4.99D 04	-2.66D 05	-4.90D 05	-1.14D 04	-1.31D 04	-2.34D 02	0.00
-2.66D 05	1.97D 06	7.95D 03	1.89D 05	2.18D 05	3.94D 03	0.00
-4.90D 02	7.95D 03	1.02D 04	4.24D 04	8.16D 03	4.96D 03	0.00
-1.14D 04	1.89D 05	4.24D 04	4.40D 05	1.97D 05	2.09D 04	0.00
-1.31D 04	2.18D 05	8.16D 03	1.97D 05	2.30D 05	4.04D 03	0.00
-2.34D 02	3.94D 03	4.96D 03	2.09D 04	4.04D 03	2.15D 04	6.42D 03
0.00	0.00	0.00	0.00	0.00	0.00	0.00
- (DA) -						
-1.35D 04	-2.31D 05	4.56D 04	-6.00D 03	1.73D 03	-9.68D 02	0.00
-2.53D 05	2.01D 05	-8.87D 05	4.95D 03	6.25D 04	-1.29D 03	0.00
1.81D 03	5.51D 02	5.75D 03	6.77D 03	6.55D 02	9.68D 03	0.00
3.66D 04	1.52D 04	-6.27D 04	8.42D 04	6.15D 04	2.01D 04	0.00
7.47D 04	2.53D 04	-4.78D 04	2.83D 04	9.52D 04	1.49D 03	0.00
-4.46D 02	7.92D 01	2.01D 04	1.26D 03	1.42D 03	2.25D 04	0.00
0.00	0.00	0.00	0.00	0.00	0.00	3.88D 03
- (DV) -						
-2.05D 11	-1.51D 11	1.16D 10	-8.87D 09	-6.36D 10	-5.12D 08	0.00
-1.90D 12	1.28D 12	-1.93D 11	1.47D 11	7.79D 11	8.40D 09	0.00
1.73D 10	8.62D 09	-8.15D 09	1.85D 10	1.85D 10	1.07D 10	0.00
4.15D 11	2.06D 11	-2.51D 11	2.53D 11	4.00D 11	4.48D 10	0.00
4.83D 11	2.39D 11	-2.03D 11	1.54D 11	4.07D 11	8.62D 09	0.00
8.59D 09	4.28D 09	9.38D 09	9.13D 09	9.16D 09	1.87D 10	0.00
0.00	0.00	0.00	0.00	0.00	0.00	1.51D 09

Figure 6.15 CONFIGURATION V - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 6
 MATRICES ARE 6X6

---- SYSM ----					
7.70D-05	1.09D-05	0.0	0.0	-6.42D-06	0.0
1.09D-05	1.97D-06	0.0	0.0	-1.48D-06	0.0
0.0	0.0	2.01D-04	-2.44D-05	1.44D-05	0.0
0.0	0.0	-2.44D-05	6.66D-06	-5.00D-06	0.0
-6.42D-06	-1.48D-06	1.44D-05	-5.00D-06	8.64D-06	0.0
0.0	0.0	0.0	0.0	1.55D-04	

---- DAMP ----					
1.32D-00	2.93D-01	5.54D-01	-1.69D-01	2.49D-01	0.0
2.93D-01	1.42D-01	1.00D-01	-3.70D-02	2.01D-02	0.0
5.54D-01	1.00D-01	1.51D-00	-3.36D-01	-5.39D-02	0.0
-1.69D-01	-3.70D-02	-3.36D-01	2.71D-01	-6.93D-02	0.0
2.49D-01	2.01D-02	-5.39D-02	-6.93D-02	4.69D-01	0.0
0.0	0.0	0.0	0.0	6.05D-01	

---- STF ----					
2.29D-06	1.15D-06	0.0	0.0	1.15D-06	0.0
1.15D-06	7.63D-05	0.0	0.0	3.82D-05	0.0
0.0	0.0	1.39D-06	-5.09D-05	-5.09D-05	0.0
0.0	0.0	-5.09D-05	5.09D-05	2.54D-05	0.0
1.15D-06	3.82D-05	-5.09D-05	2.54D-05	1.27D-06	0.0
0.0	0.0	0.0	0.0	2.36D-05	

Figure 6.16 CONFIGURATION VI - Mass, Damping, and Stiffness

CONFIGURATION AND MATRIX LAYER = 6
MATRICES ARE 6X1

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
0.0	0.0	0.0
7.77D-05	-6.05D-01	-2.36D 05

Figure 6.17 CONFIGURATION VI - Precalculated Matrices; COLM, COLC, and COLK

CONFIGURATION = 6 MATRIX LAYER = 1
MATRICES ARE 6x1

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
0.0	-7.00D-02	0.0
0.0	-1.01D-02	0.0
-2.59D-05	9.12D-01	7.07D 05
0.0	-1.09D-01	0.0
0.0	5.66D-02	0.0
0.0	0.0	0.0

Figure 6.13 CONFIGURATION VI - Precalculated Matrices; COLM, COLC, and COLK

CONFIGURATION AND MATRIX LAYER = 6
MATRICES ARE 6X6

- (BMIV) -					
4.99D 04	-2.66D 05	-4.36D 02	-1.11D 04	-1.31D 04	0.00
-2.66D 05	1.97D 06	7.04D 03	1.85D 05	1.17D 05	0.00
-4.36D 02	7.04D 03	9.03D 03	3.76D 04	7.23D 03	0.00
-1.11D 04	1.85D 05	3.76D 04	4.20D 05	1.93D 05	0.00
-1.31D 04	2.17D 05	7.23D 03	1.93D 05	2.29D 05	0.00
0.00	0.00	0.00	0.00	6.42D 03	
- (DA) -					
-1.35D 04	-2.31D 04	4.78D 03	-5.86D 02	1.74D 03	0.00
-2.53D 05	2.01D 05	-1.26D 04	4.71D 03	6.22D 04	0.00
1.91D 03	5.32D 02	-1.10D 03	6.48D 03	3.28D 02	0.00
3.70D 04	1.51D 04	-8.22D 04	8.30D 04	6.01D 04	0.00
7.48D 04	2.53D 04	-5.16D 04	2.81D 04	9.49D 04	0.00
0.00	0.00	0.00	0.00	3.88D 03	
- (DV) -					
-2.05D 11	-1.51D 11	1.17D 10	-8.77D 09	-6.35D 10	0.00
1.90D 12	-1.28D 12	-1.95D 11	1.46D 11	7.68D 11	0.00
1.53D 10	7.64D 09	-1.03D 10	1.64D 10	1.64D 11	0.00
4.07D 11	2.02D 11	-2.60D 11	2.44D 11	3.91D 11	0.00
4.82D 11	2.39D 11	-2.05D 11	1.53D 11	4.05D 11	0.00
0.00	0.00	0.00	0.00	1.51D 09	

Figure 6.19 CONFIGURATION VI - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 7
 MATRICES ARE 6X6

--- SYSM ---					
--- DAMP ---					
--- STF ---					
2•32D-04	1•09D-05	0•0	0•0	-6•42D-06	0•0
1•09D-05	1•97D-06	0•0	0•0	-1•48D-06	0•0
0•0	0•0	2•01D-04	-2•44D-05	-1•44D-05	-2•59D-05
0•0	0•0	-2•44D-05	6•66D-06	-5•00D-06	0•0
-6•42D-06	-1•48D-06	-1•44D-05	-5•00D-06	8•64D-06	0•0
0•0	0•0	-2•59D-05	0•0	0•0	5•18D-05
3•49D 00	3•58D-01	2•09D 00	-3•08D-01	3•16D-01	-6•69D-01
3•58D-01	2•03D-01	1•02D-01	-3•48D-02	2•59D-03	-1•65D-02
2•09D 00	1•02D-01	3•10D 00	-4•83D-01	3•25D-02	-2•99D-01
-3•08D-01	-3•48D-02	-4•83D-01	2•86D-01	-8•02D-02	-5•40D-02
-3•16D-01	2•59D-03	3•25D-02	-8•02D-02	4•91D-01	2•90D-02
-6•69D-01	-1•65D-02	2•99D-01	-5•40D-02	2•90D-02	1•17D 00

Figure 6.20 CONFIGURATION VII - Mass, Damping, and Stiffness

CONFIGURATION AND MATRIX LAYER = 7
MATRICES ARE 6X1

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
7.77D-05	1.05D 00	-2.36D 05
0.0	-7.37D-03	0.0
0.0	1.72D 00	0.0
0.0	-1.60D-01	0.0
0.0	9.80D-02	0.0
0.0	-6.62D-01	0.0

Figure 6.21 CONFIGURATION VII - Pre-calculated Matrices; COLM, COLC, and COLK

CONFIGURATION AND MATRIX LAYER = 7

MATRICES ARE 6X6

- (BMIV) -

5.65D	03	-2.97D	04	-7.29D	01	-1.24D	03	-1.42D	03	-4.55D	00
-2.97D	04	7.05D	05	5.65D	03	1.33D	05	1.54D	05	2.71D	03
-7.29D	01	5.65D	03	1.01D	04	4.23D	04	8.03D	03	4.97D	03
-1.24D	03	1.33D	05	4.23D	04	4.38D	05	1.93D	05	2.08D	04
-1.42D	03	1.54D	05	8.03D	03	1.93D	05	2.27D	05	3.97D	03
-4.55D	00	2.71D	03	4.97D	03	2.08D	04	3.97D	03	2.15D	04

- (DA) -

8.83D	03	-3.98D	03	9.07D	03	-9.13D	02	1.10D	03	-3.29D	03
1.67D	05	1.29D	05	-3.08D	04	7.44D	03	5.77D	04	1.04D	04
9.12D	03	6.28D	02	-1.32D	04	6.10D	03	1.01D	03	6.76D	03
4.39D	04	1.59D	04	-5.67D	04	8.37D	04	6.17D	04	1.77D	04
7.63D	04	2.54D	04	-4.72D	04	2.81D	04	9.60D	04	1.59D	03
-8.21D	03	-1.21D	01	1.22D	04	1.98D	03	1.07D	03	2.56D	04

- (DV) -

-2.14D	10	-1.68D	10	1.25D	09	-9.58D	08	-6.96D	09	-5.48D	07
-9.09D	11	5.63D	11	-1.37D	11	1.04D	11	4.62D	11	5.91D	09
1.55D	10	7.29D	09	-8.04D	09	1.84D	10	1.79D	10	1.07D	10
3.71D	11	1.74D	11	-2.48D	11	2.50D	11	3.85D	11	4.46D	10
4.32D	11	2.03D	11	-2.00D	11	1.52D	11	3.91D	11	8.48D	09
7.64D	09	3.58D	09	9.44D	09	9.09D	09	8.85D	09	1.87D	10

Figure 6.22 CONFIGURATION VII - Precalculated Matrices; BMIV, DA, and DV

CONFIGURATION AND MATRIX LAYER = 8
 MATRICES ARE 5X5

--- SYSM ---

2.32D-04	1.09D-05	0.0	0.0	-6.42D-06
1.09D-05	1.97D-06	0.0	0.0	-1.48D-06
0.0	0.0	2.01D-04	-2.44D-05	-1.44D-05
0.0	0.0	-2.44D-05	6.66D-06	-5.00D-06
-6.42D-06	-1.48D-06	1.44D-05	-5.00D-06	8.64D-06

--- DAMP ---

3.49D 00	3.58D-01	2.09D 00	-3.08D-01	3.16D-01
3.58D-01	2.03D-01	1.02D-01	-3.48D-02	2.59D-03
2.09D 00	1.02D-01	3.10D 00	-4.83D-01	3.25D-02
-3.08D-01	-3.48D-02	-4.83D-01	2.86D-01	-8.02D-02
-3.16D-01	2.59D-03	3.25D-02	-8.02D-02	4.91D-01

--- STF ---

2.53D 06	1.15D 06	0.0	0.0	1.15D 06
1.15D 06	7.63D 05	0.0	0.0	3.82D 05
0.0	0.0	1.39D 06	-5.09D 05	-5.09D 05
0.0	0.0	-5.09D 05	5.09D 05	2.54D 05
1.15D 06	3.82D 05	-5.09D 05	2.54D 05	1.27D 06

Figure 6.23 CONFIGURATION VIII - Mass, Damping, and Stiffness

CONFIGURATION AND MATRIX LAYER = 8
MATRICES ARE 5X1

COLUMN MATRICES FOR CALCULATION OF DELTA P

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
7.77D-05	1.05D 00	-2.36D 05
0.0	-7.37D-03	0.0
0.0	1.72D 00	0.0
0.0	-1.60D-01	0.0
0.0	9.80D-02	0.0

CONFIGURATION = 8, MATRIX LAYER = 3
MATRICES ARE 5X1

COLUMN MATRICES FOR CALCULATION OF DELTA P

COLUMN MASS	COLUMN DAMPING	COLUMN STIFFNESS
0.0	-6.69D-01	0.0
0.0	-1.65D-02	0.0
-2.59D-05	2.99D-01	7.07D 05
0.0	-5.40D-02	0.0
0.0	2.90D-02	0.0

Figure 6.24 CONFIGURATION VIII - Precalculated Matrices; COLM, COLC, and COLK

CONFIGURATION AND MATRIX LAYER = 8
MATRICES ARE 5X5

- (BMIV) -

5.65D	03	-2.97D	04	-7.19D	01	-1.24D	03	-1.42D	03
-2.97D	04	7.05D	05	5.02D	03	1.31D	05	1.54D	05
-7.19D	01	5.02D	03	9.00D	03	3.75D	04	7.11D	03
-1.24D	03	1.31D	05	3.75D	04	4.18D	05	1.90D	05
-1.42D	03	1.54D	05	7.11D	03	1.90D	C5	2.26D	05

- (DA) -

8.83D	03	-3.98D	03	9.07D	03	-9.12D	02	1.10D	03
1.68D	05	1.29D	05	-3.23D	04	7.19D	03	5.75D	04
1.10D	04	6.31D	02	-1.04D	04	5.64D	03	7.64D	02
5.19D	04	1.59D	04	-6.85D	04	8.18D	04	6.07D	04
7.78D	04	2.54D	04	-4.95D	04	2.77D	04	9.58D	04

- (DV) -

-2.14D	10	-1.68D	10	1.26D	09	-9.56D	08	-6.96D	09
-9.08D	11	5.63D	11	-1.39D	11	1.03D	11	4.61D	11
1.37D	10	6.47D	09	-1.02D	10	1.63D	10	1.58D	10
3.64D	11	1.71D	11	-2.57D	11	2.42D	11	3.77D	11
4.31D	11	2.02D	11	-2.02D	11	1.50D	11	3.89D	11

Figure 6.25 CONFIGURATION VIII - Precalculated Matrices; BMIV, DA, and DV

C. SAMPLE PROBLEMS

One set of results for 11,000 RPM was obtained in CALCOMP graph form and as AGT graphic display photographs. At this high speed of cam rotation, the pushrod is frequently thrown off the cam. The results are completely analyzed with numerous reference points plotted on the graphs.

The second example problem was chosen at a slower speed, 9,000 RPM. At this speed the pushrod remains in contact with the cam. A third intermediate example, 10,000 RPM was chosen for a comparison between the two above examples. Figure 6.26 shows the relationship between the sample problems.

Photographs from the AGT display are presented after the CALCOMP graphs for each case of example one. It is noted that large scale factors are introduced for both bending and axial deformation as an aid to the user.

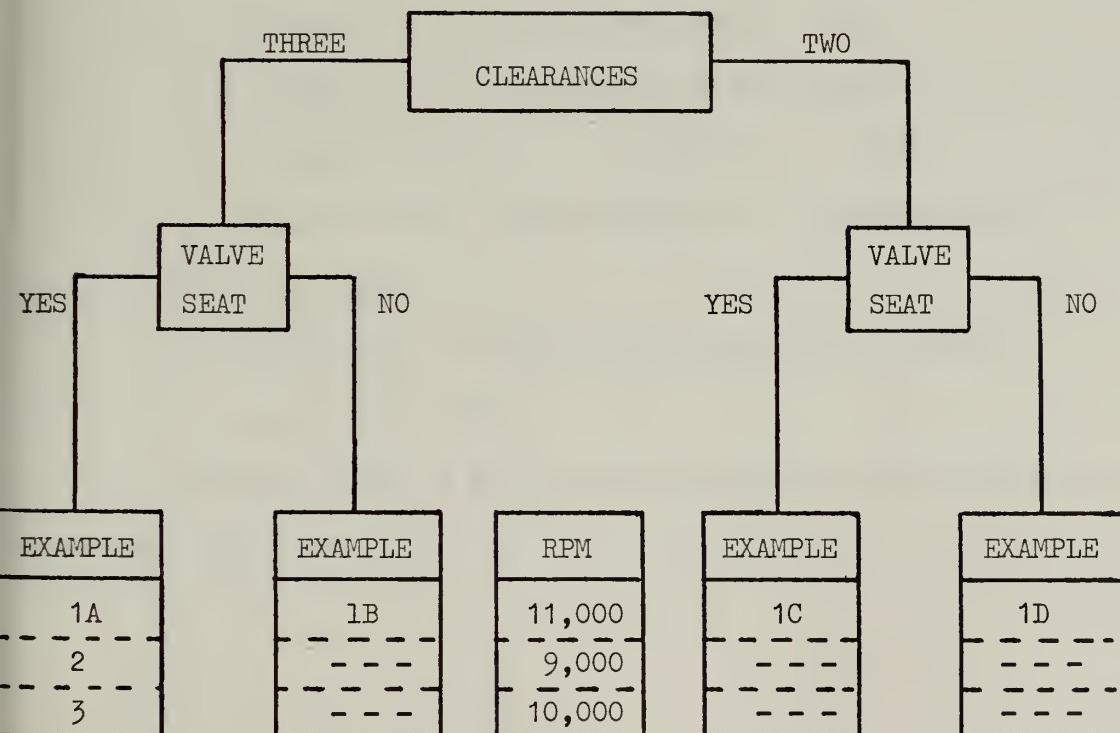


Figure 6.26 SAMPLE NUMERICAL RESULTS

1. Example 1A, Three Clearance, 11,000 RPM, Valve Seat

This analysis will commence at 270 degrees of cam rotation.

One cycle will be discussed, from point (A) to point (AA). The following figures are utilized for the analysis:

a. Figure 6.27

(1) Displacement difference between cam and pushrod

$$(q_1 - q_9) \times 10^{-2}$$

(2) Displacement of lower end of pushrod i.e. follower,

$$q_1 \times 10^{-1}$$

(3) Cam Profile $q_9 \times 10^{-1}$

b. Figure 6.28

(1) Displacement difference between pushrod and rocker

$$(q_2 - q_8) \times 10^{-2}$$

(2) Displacement difference between cam and pushrod

$$(q_1 - q_9) \times 10^{-2}$$

c. Figure 6.29 Force on the Pushrod $F_1 \times 10^3$

d. Figure 6.30 Force on the Rocker $F_2 \times 10^2$

e. Figure 6.31 Displacement of the Valve $q_7 \times 10^{-1}$

f. Figure 6.32 Force on the Valve $F_7 \times 10^3$

g. Figure 6.33 Axial Oscillation of the Valve Stem

$$(q_7 - q_6) \times 10^{-3}$$

h. Figure 6.34 Axial Oscillations of the Pushrod

$$(q_8 - q_1) \times 10^{-3}$$

Figures 6.33 and 6.34 have been included for completeness and are not annotated.

The following reference points approximately correspond to the indicated degrees of cam rotation, Table 6.1

Reference Point	Degrees of Cam Rotation	Reference Point	Degrees of Cam Rotation
A	270	I	380
B	288	J	418
C	294	K	430
D	300	L	448
E	336	M	459
F	346	N	579
G	357	P	583
H	363	AA	630

Table 6.1 Graph Reference Points and Degrees of Cam Rotation for Figures 6.27 - 6.32

At point (A), the cam is at its lowest extreme and is starting to rise. The pushrod is floating between the cam and rocker, with downward motion. The valve has just undergone a severe oscillation and has resumed its upward motion.

At point (B), the cam and pushrod impact and remain in contact for a short period, (B-C). The rising cam and force of impact create severe oscillation of the pushrod and a bounce occurs. The pushrod is again thrown clear of the cam and floats (C-E).

At point (D), the valve has bounced against the valve seat, creating severe oscillation and more bouncing of the valve (D-E). The pushrod is floating thus is not affected by this impact.

At point (E), the rising pushrod now impacts against the rocker, and valve against the valve seat. At point (E), the valve has again reseated and the rocker has become almost stationary. This allows the rising pushrod to impact against the rocker.

The pushrod bounces off the rocker (F-H) and again floats (F-H), now having downward motion and also creating downward motion of the valve.

The rising cam almost immediately contacts the pushrod (G). The resulting motion creates positive contact between cam, pushrod, and rocker (H-I), and the valve bounces cease and it resumes its downward harmonic motion. At point (I), the speed of the cam throws the pushrod clear. Immediately separation occurs at the rocker-pushrod due to inertial effects. Several bounces at this clearance are observed (J-K-L-M), until the system reaches its extreme, point (M), i.e. the cam now with downward motion begins falling away from the pushrod. The valve spring, having reached the maximum extension, reverses the motion of the valve upward. The rocker and pushrod are now in contact (M) and both have downward motion.

At point (N), the valve impacts against the seat which results in severe oscillations and valve bounce (P-AA). With the gross motion of the rocker halted, the pushrod floats clear (P-AA). The cycle then repeats.

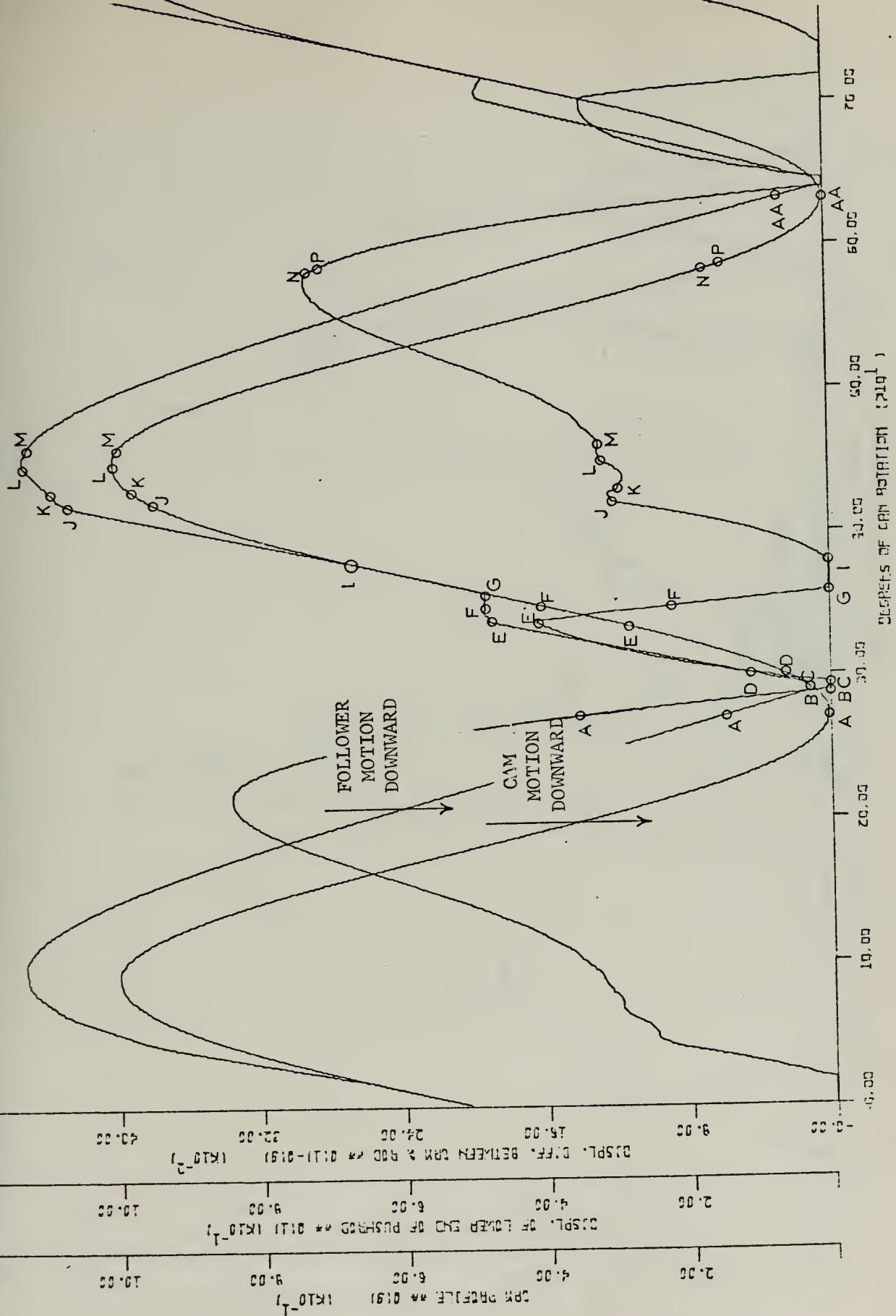


Figure 6.27 THREE CLEARANCES

11,000 RPM

L VALUE SEAT

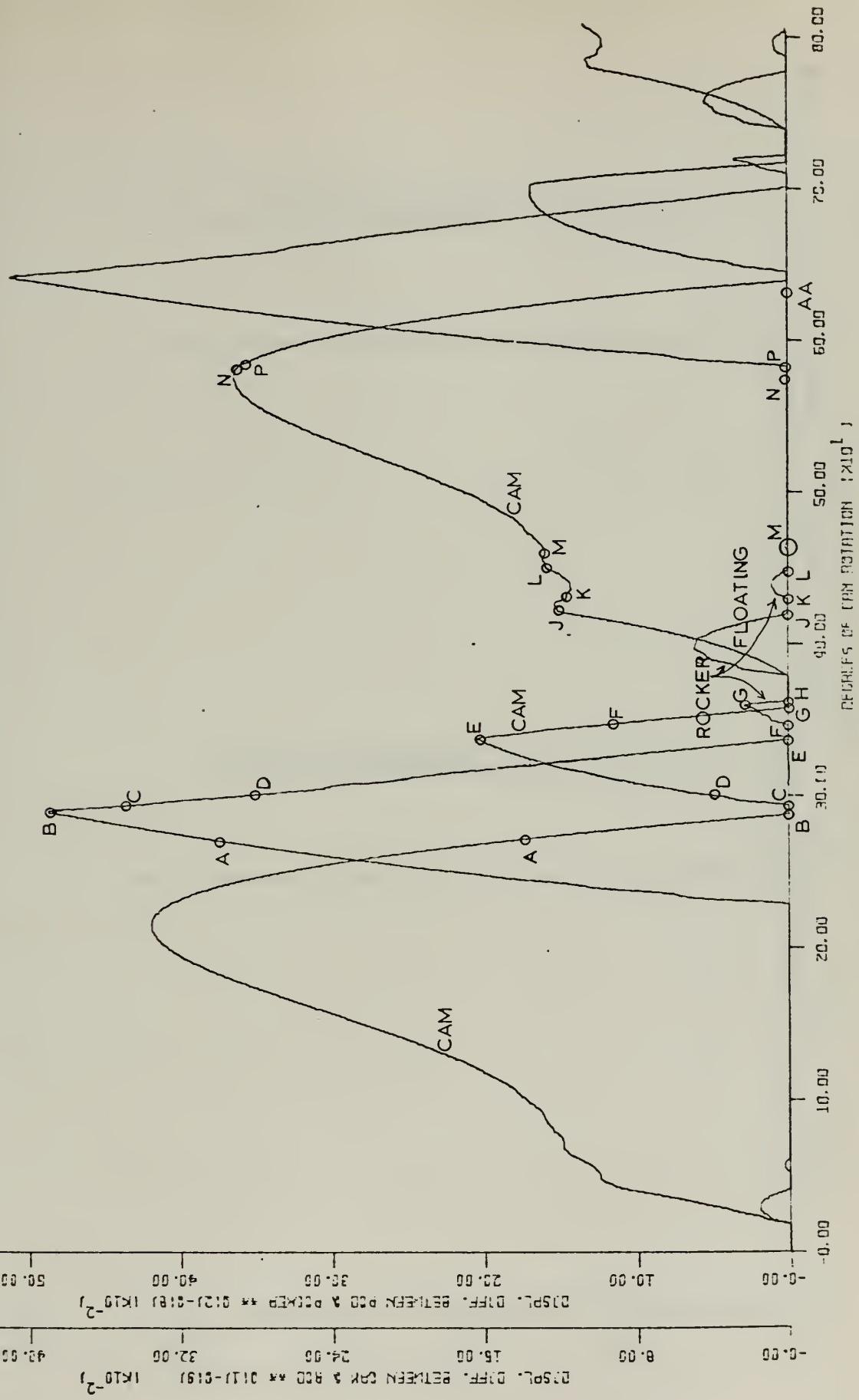
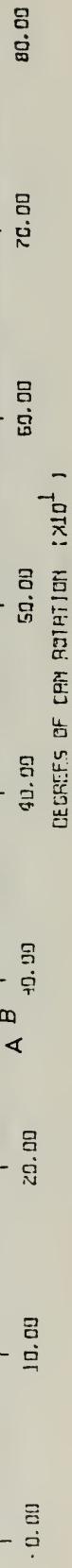


Figure 6.28 THREE CLEARANCES 11,000 RPM VALVE SEAT

WHL UE SEAL

11,000 RPM

Figure 6.29 THREE CLEARANCES



11. DUO RPM VALVE SEAT

Figure 6.30 THREE CLEARANCES

Degrees of cam rotation (deg°)

-0.00 10.00 20.00 30.00 40.00 50.00 60.00

60.00

50.00

40.00

30.00

20.00

10.00

0.00

-10.00

-20.00

-30.00

-40.00

-50.00

-60.00

-70.00

-80.00

-90.00

-100.00

-110.00

-120.00

-130.00

-140.00

-150.00

-160.00

-170.00

-180.00

-190.00

-200.00

-210.00

-220.00

-230.00

-240.00

-250.00

-260.00

-270.00

-280.00

-290.00

-300.00

-310.00

-320.00

-330.00

-340.00

-350.00

-360.00

-370.00

-380.00

-390.00

-400.00

-410.00

-420.00

-430.00

-440.00

-450.00

-460.00

-470.00

-480.00

-490.00

-500.00

-510.00

-520.00

-530.00

-540.00

-550.00

-560.00

-570.00

-580.00

-590.00

-600.00

-610.00

-620.00

-630.00

-640.00

-650.00

-660.00

-670.00

-680.00

-690.00

-700.00

-710.00

-720.00

-730.00

-740.00

-750.00

-760.00

-770.00

-780.00

-790.00

-800.00

-810.00

-820.00

-830.00

-840.00

-850.00

-860.00

-870.00

-880.00

-890.00

-900.00

-910.00

-920.00

-930.00

-940.00

-950.00

-960.00

-970.00

-980.00

-990.00

-1000.00

-1010.00

-1020.00

-1030.00

-1040.00

-1050.00

-1060.00

-1070.00

-1080.00

-1090.00

-1100.00

-1110.00

-1120.00

-1130.00

-1140.00

-1150.00

-1160.00

-1170.00

-1180.00

-1190.00

-1200.00

-1210.00

-1220.00

-1230.00

-1240.00

-1250.00

-1260.00

-1270.00

-1280.00

-1290.00

-1300.00

-1310.00

-1320.00

-1330.00

-1340.00

-1350.00

-1360.00

-1370.00

-1380.00

-1390.00

-1400.00

-1410.00

-1420.00

-1430.00

-1440.00

-1450.00

-1460.00

-1470.00

-1480.00

-1490.00

-1500.00

-1510.00

-1520.00

-1530.00

-1540.00

-1550.00

-1560.00

-1570.00

-1580.00

-1590.00

-1600.00

-1610.00

-1620.00

-1630.00

-1640.00

-1650.00

-1660.00

-1670.00

-1680.00

-1690.00

-1700.00

-1710.00

-1720.00

-1730.00

-1740.00

-1750.00

-1760.00

-1770.00

-1780.00

-1790.00

-1800.00

-1810.00

-1820.00

-1830.00

-1840.00

-1850.00

-1860.00

-1870.00

-1880.00

-1890.00

-1900.00

-1910.00

-1920.00

-1930.00

-1940.00

-1950.00

-1960.00

-1970.00

-1980.00

-1990.00

-2000.00

-2010.00

-2020.00

-2030.00

-2040.00

-2050.00

-2060.00

-2070.00

-2080.00

-2090.00

-2100.00

-2110.00

-2120.00

-2130.00

-2140.00

-2150.00

-2160.00

-2170.00

-2180.00

-2190.00

-2200.00

-2210.00

-2220.00

-2230.00

-2240.00

-2250.00

-2260.00

-2270.00

-2280.00

-2290.00

-2300.00

-2310.00

-2320.00

-2330.00

-2340.00

-2350.00

-2360.00

-2370.00

-2380.00

-2390.00

-2400.00

-2410.00

-2420.00

-2430.00

-2440.00

-2450.00

-2460.00

-2470.00

-2480.00

-2490.00

-2500.00

-2510.00

-2520.00

-2530.00

-2540.00

-2550.00

-2560.00

-2570.00

-2580.00

-2590.00

-2600.00

-2610.00

-2620.00

-2630.00

-2640.00

-2650.00

-2660.00

-2670.00

-2680.00

-2690.00

-2700.00

-2710.00

-2720.00

-2730.00

-274

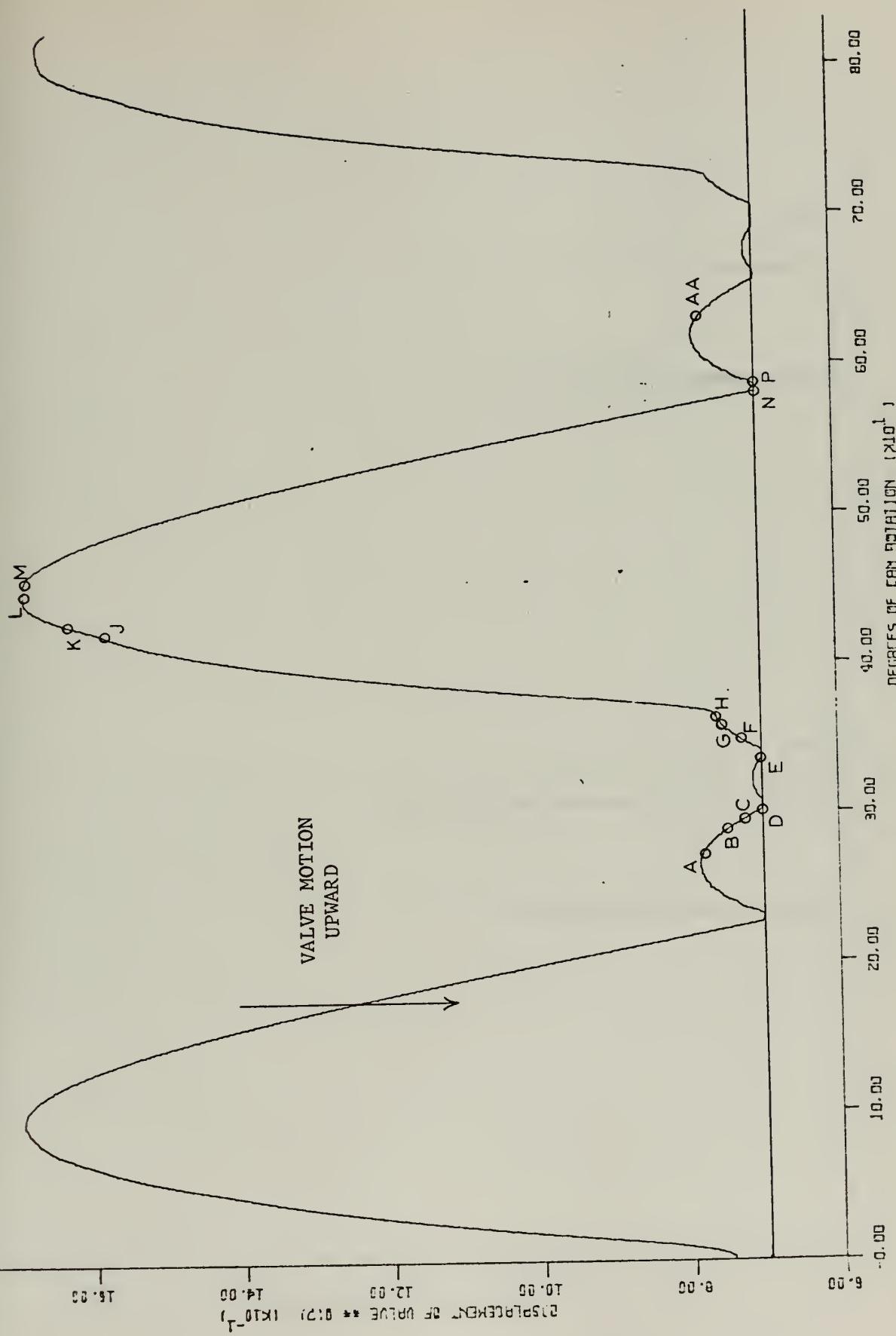
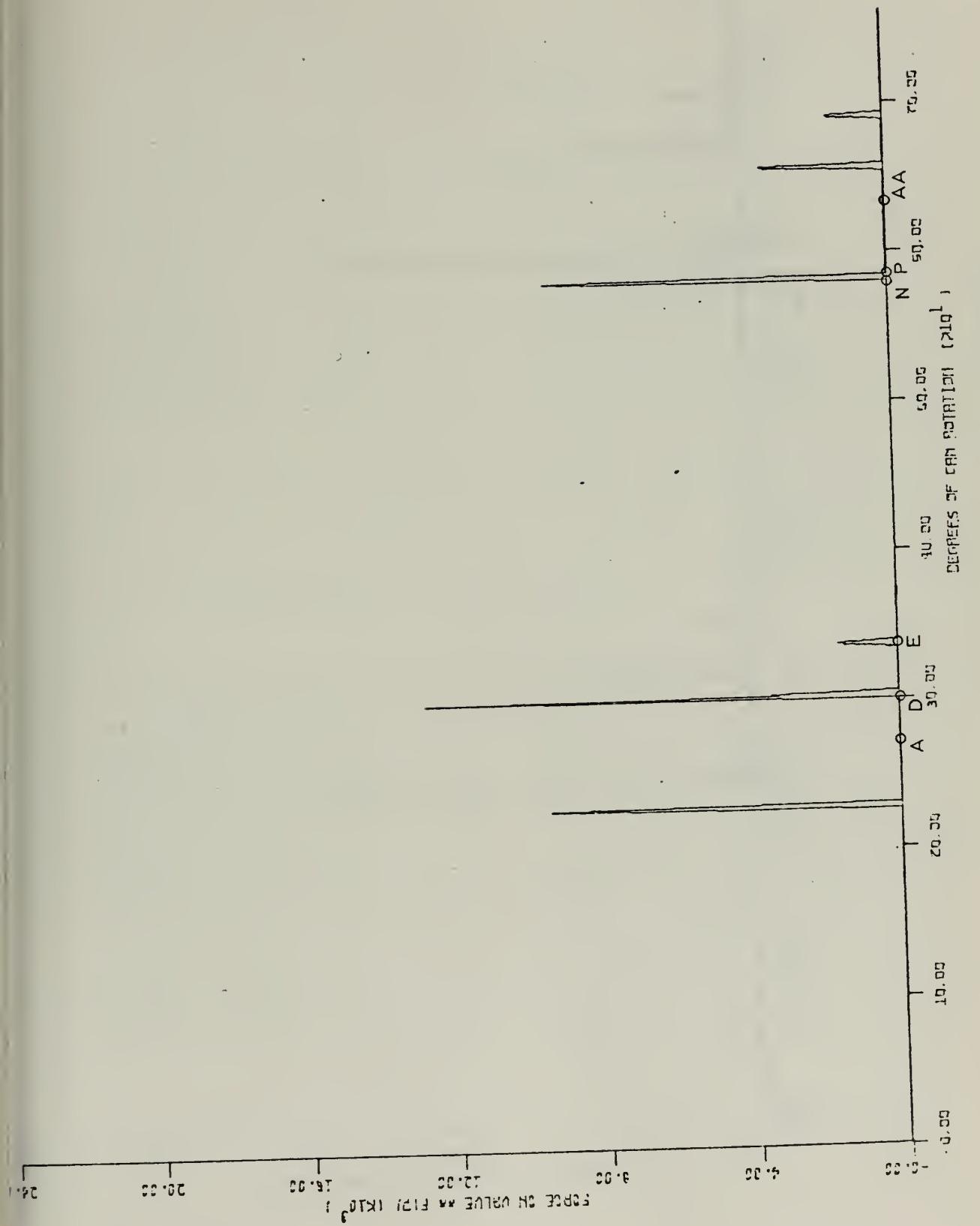


Figure 6.31 THREE CLEARANCES

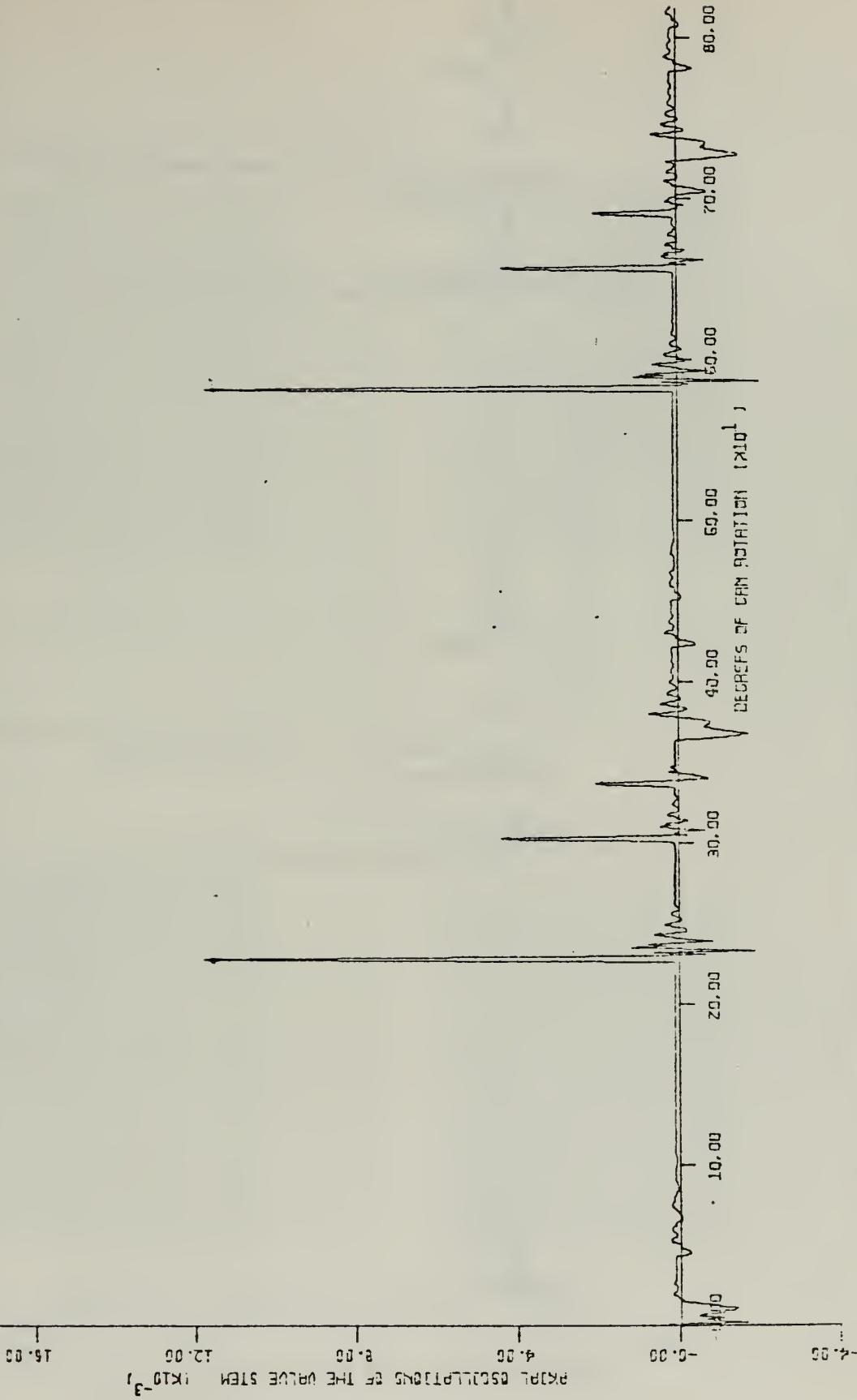
Figure 6.32 THREE CLEARANCES
11,000 RPM VALUE SEAT



LIFELINE SHIFT

11,000 RPM

THREE CLEARANCES



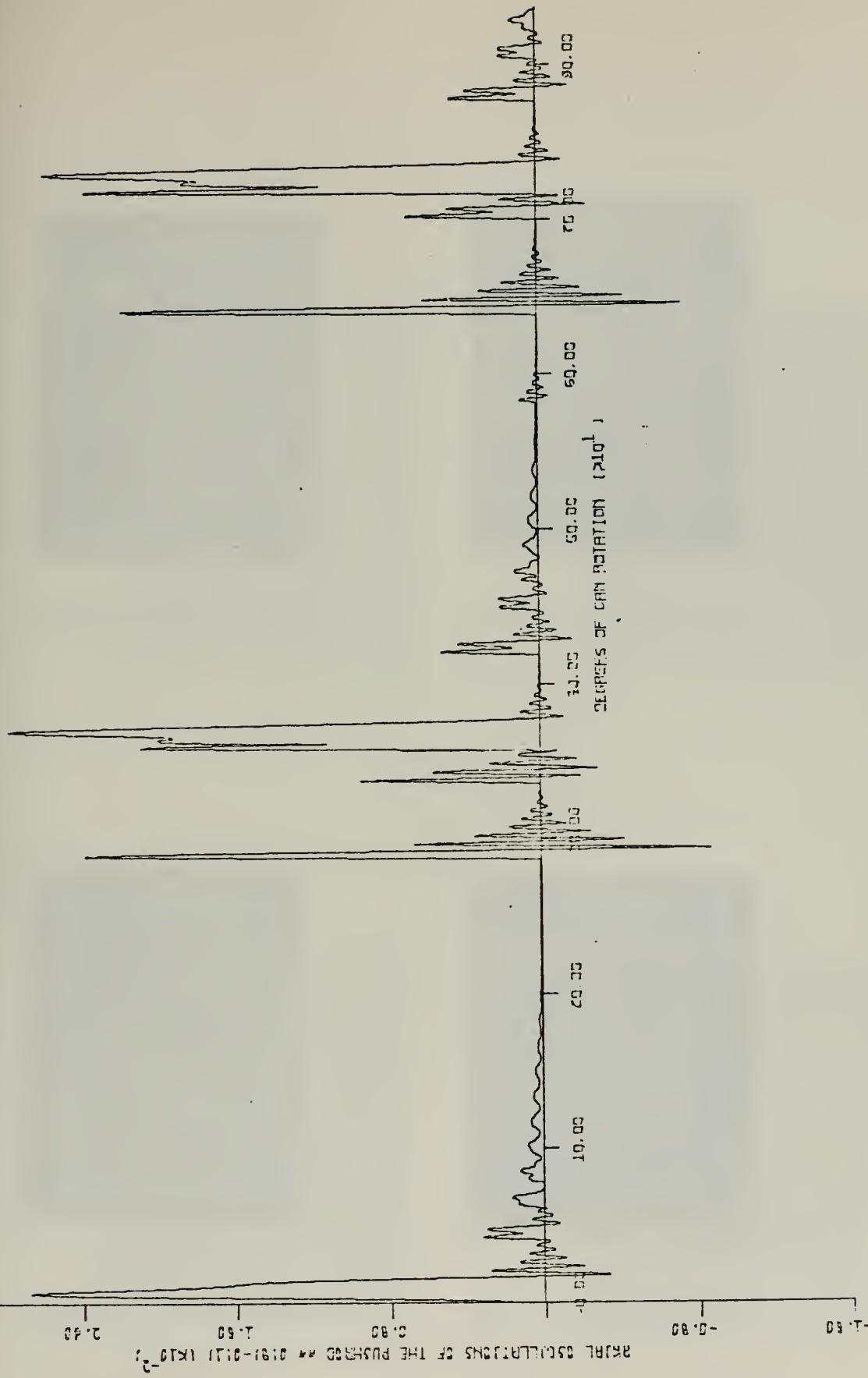


Figure 6.34 THREE CLEARANCES 11,000 RPM VALVE SEAT

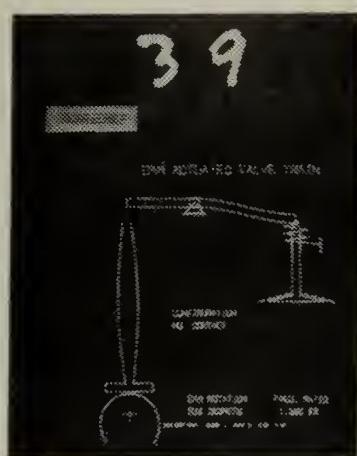


Photo 6.2 - 630 degrees

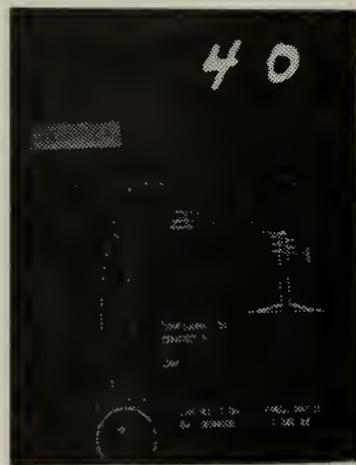


Photo 6.3 - 641 degrees

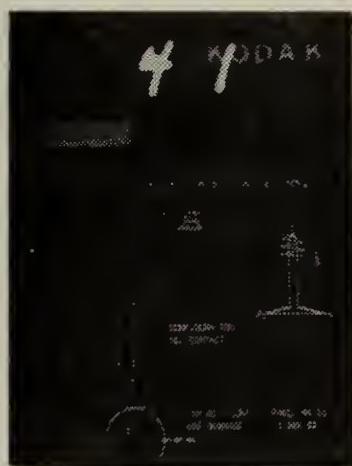


Photo 6.4 - 650 degrees

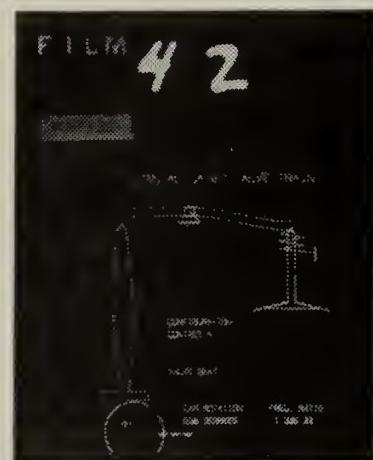


Photo 6.5 - 658 degrees



Photo 6.6 - 664 degrees

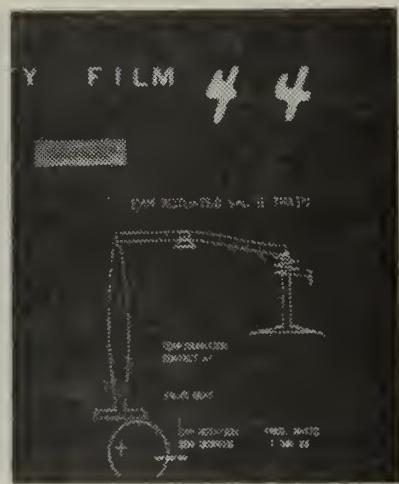


Photo 6.7 - 682 degrees

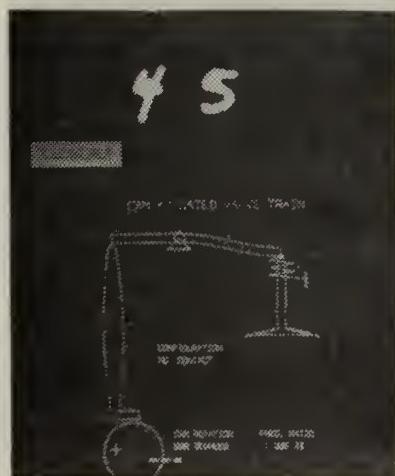


Photo 6.8 - 686 degrees

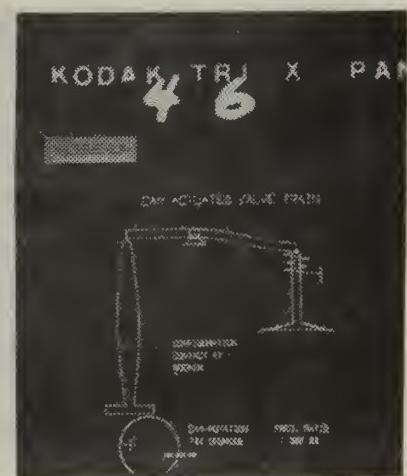


Photo 6.9 - 704 degrees

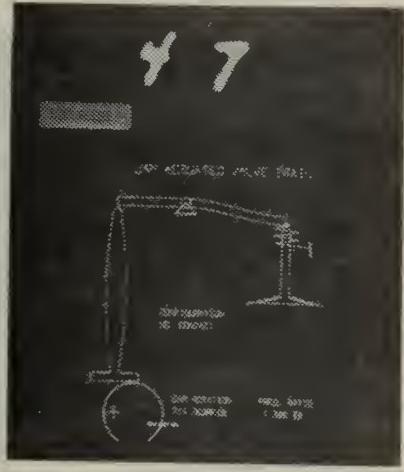


Photo 6.10 - 71 $\frac{1}{4}$ degrees

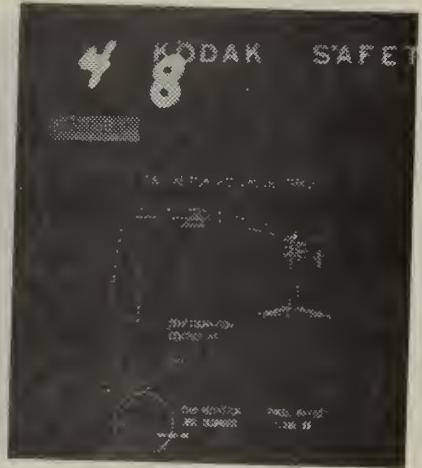


Photo 6.11 - 362 degrees

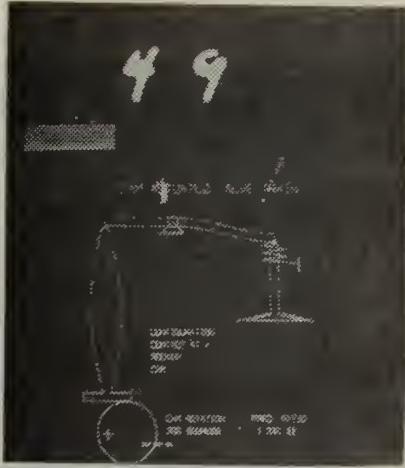


Photo 6.12 - 366 degrees

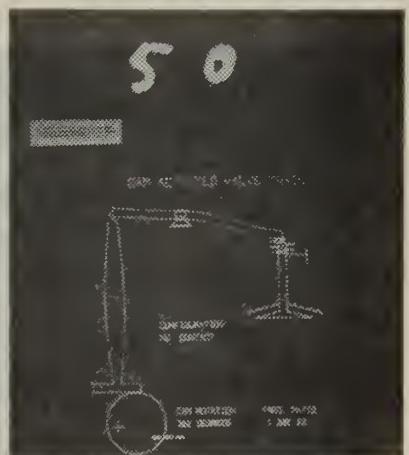


Photo 6.13 - 380 degrees

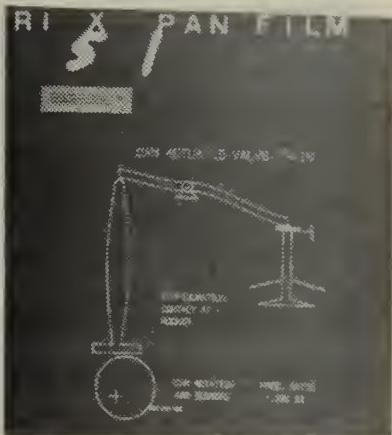


Photo 6.14 - 422 degrees

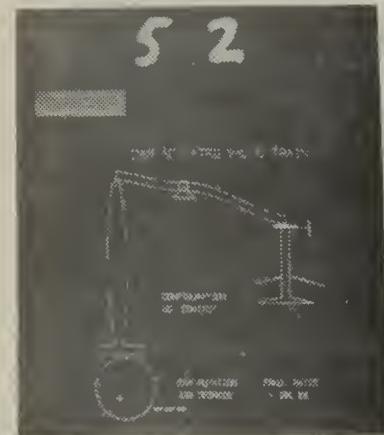


Photo 6.15 - 432 degrees

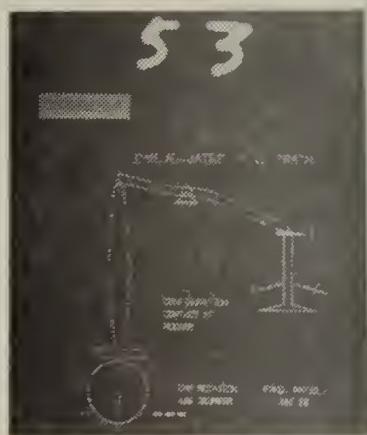


Photo 6.16 - 456 degrees

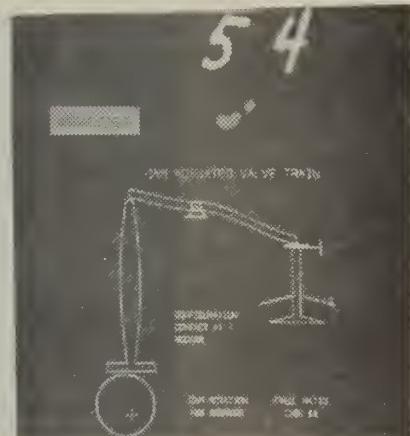


Photo 6.17 - 468 degrees

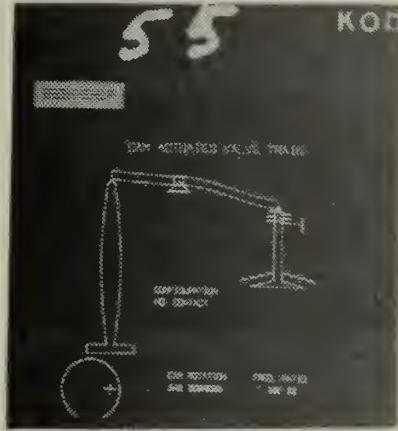


Photo 6.18 - 546 degrees

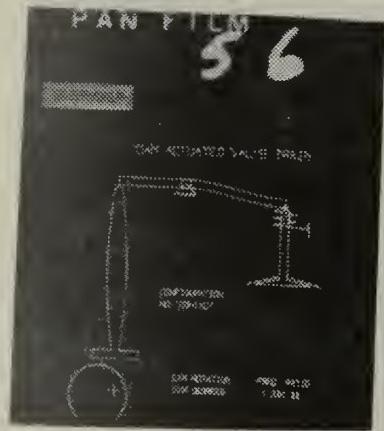


Photo 6.19 - 572 degrees

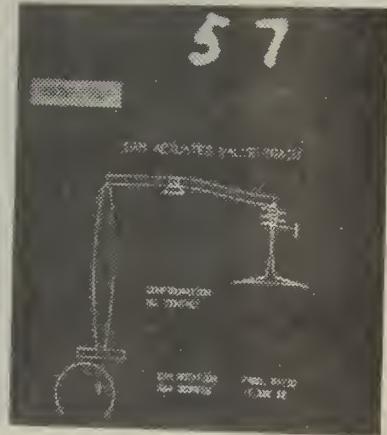


Photo 6.20 - 582 degrees



Photo 6.21 - 58⁴ degrees

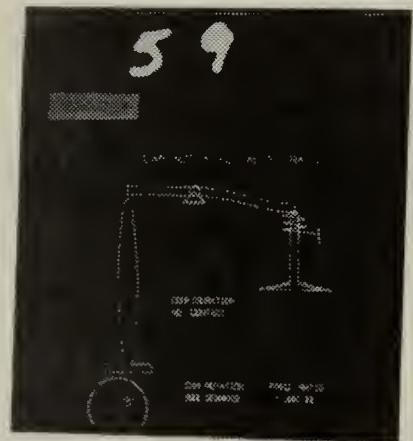


Photo 6.22 - 600 degrees

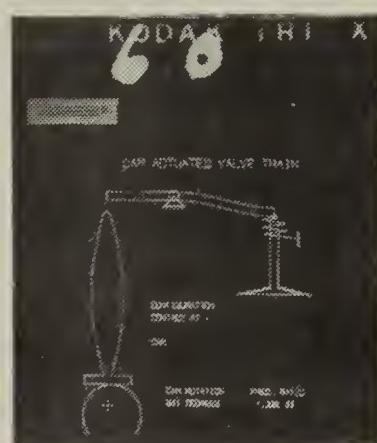


Photo 6.23 - 641 degrees

2. Example 1B, Three Clearances, 11,000 RPM, No Valve Seat

This analysis starts at 274 degrees of cam rotation, point

(A). The following figures are utilized for the analysis.

a. Figure 6.35

(1) Displacement difference between cam and pushrod

$$(q_1 - q_9) \times 10^{-2}$$

(2) Displacement of lower end of pushrod, i.e. follower

$$q_1 \times 10^{-1}$$

(3) Cam profile, $q_9 \times 10^{-1}$

b. Figure 6.36

(1) Displacement difference between pushrod and rocker

$$(q_2 - q_8) \times 10^{-2}$$

(2) Displacement difference between cam and pushrod

$$(q_1 - q_9) \times 10^{-2}$$

c. Figure 6.37 Forces on the pushrod, $F_1 \times 10^3$

d. Figure 6.38 Forces on the rocker, $F_2 \times 10^2$

e. Figure 6.39 Displacement of Valve, $q_7 \times 10^2$

f. Figure 6.40 Axial oscillation of the pushrod,

$$(q_8 - q_1) \times 10^{-3}$$

Figure 6.40 has been included for completeness and is not annotated.

The following reference points approximately correspond to the indicated degrees of cam rotation, Table 6.2

Reference Point	Degrees of Cam Rotation	Reference Point	Degrees of Cam Rotation
A	274	H	401
B	288	I	413
C	295	J	419
D	314	K	449
E	358	L	581
F	364	AA	634
G	382		

Table 6.2 Graph Reference Points and Degrees of Cam Rotation For Figures 6.35-6.39.

The analysis is similar to example 1A, except here the motion of the system is not impeded by the valve seat.

At point (A), the cam is at its lowest extreme and is starting to rise. The pushrod is floating between the cam and rocker, with downward motion.

At point (B), the cam and pushrod impact against one another and remain in contact for a short period, (B-C). The rising cam and force of impact create severe oscillations of the pushrod and a bounce occurs. The system remains intact, i.e. contact at the cam and rocker, (C-D). The valve has reached its highest extreme and reverses motion downwards.

At point (D), the pushrod and cam separate. Immediately, contact is also lost at the rocker and the pushrod floats free, (D-E).

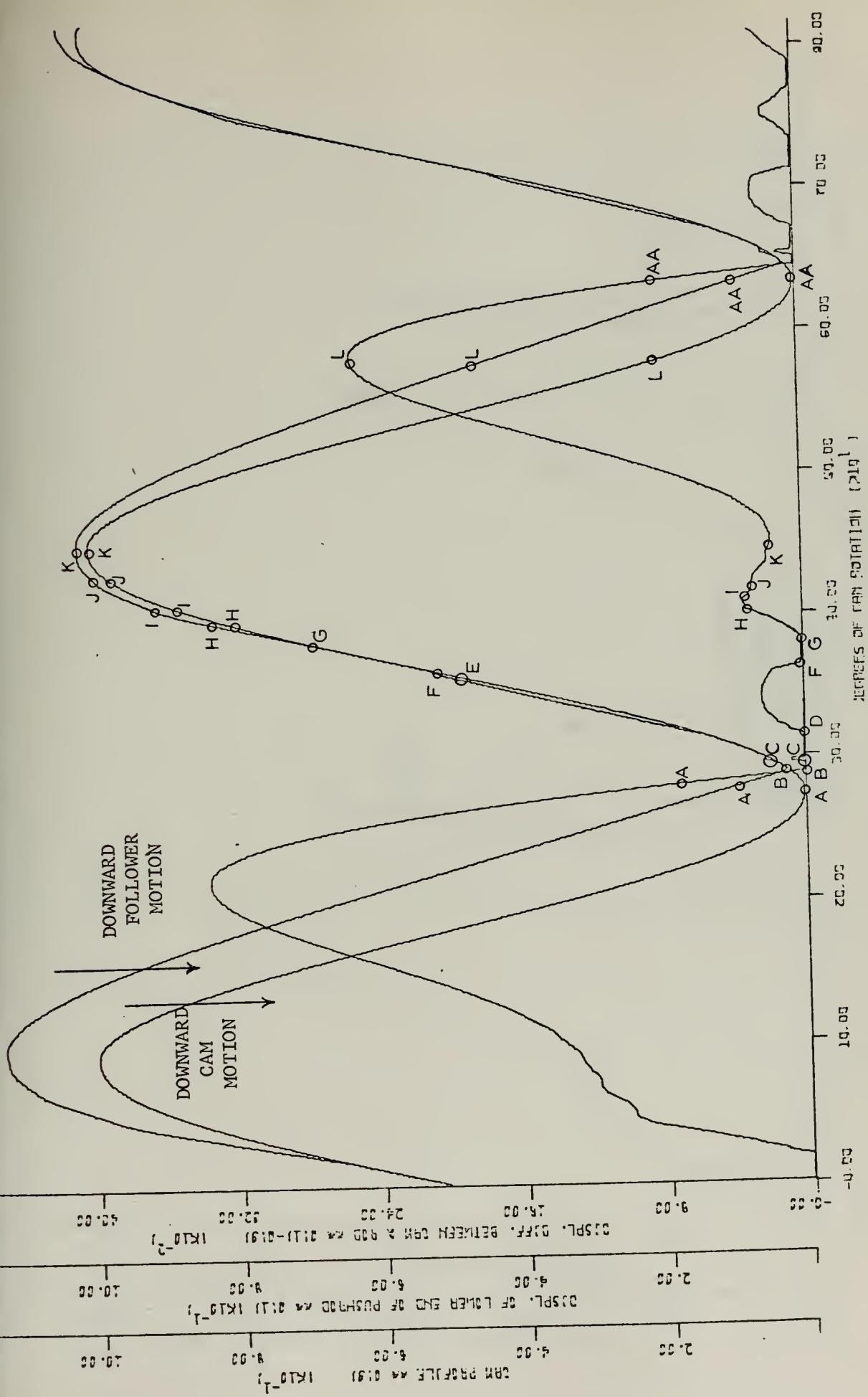
At point (E), the rising pushrod now impacts against the rocker. This slows the motion of the pushrod sufficiently for the rising cam to again impact against the rocker, point (F). The system then remains in contact for a short period (F-G).

At point (G), the speed of the cam throws the pushrod clear. Immediately separation also occurs at the rocker - pushrod due to inertial effects. The pushrod floats free for a short time, (G-H), and then bounces at the rocker-pushrod clearance, point (I). The system next reaches its extreme value, point (K), i.e. the cam now with downward motion begins falling away from the pushrod. The valve spring, having reached the maximum extension, reverses the harmonic motion of the valve, upward. The rocker and pushrod are now in contact (J), and both have downward harmonic motion.

Finally, the compression of the valve spring slows the rocker valve assembly until the pushrod loses contact at the rocker and again floats free, point (L).

At point (AA), 634 degrees of cam rotation, the cycle then repeats.

Figure 6.35 THREE CLEARANCES 111,000 RPM NO VALUE SEAT



11,000 RPM NO VALVE SEAT

Figure 6.36 THREE CLEARANCES

DEGREES OF FIN ROTATION (10^{-1})

-0.00 10.00 20.00 30.00 40.00 50.00 60.00 70.00 80.00

80.00

70.00

60.00

50.00

40.00

30.00

20.00

10.00

0.00

-0.00

22.0

10.0

0.0

-2.0

-4.0

-6.0

-8.0

-10.0

-12.0

-14.0

-16.0

-18.0

-20.0

-22.0

-24.0

-26.0

-28.0

-30.0

-32.0

-34.0

-36.0

-38.0

-40.0

-42.0

-44.0

-46.0

-48.0

-50.0

-52.0

-54.0

-56.0

-58.0

-60.0

-62.0

-64.0

-66.0

-68.0

-70.0

-72.0

-74.0

-76.0

-78.0

-80.0

-82.0

-84.0

-86.0

-88.0

-90.0

-92.0

-94.0

-96.0

-98.0

-100.0

-102.0

-104.0

-106.0

-108.0

-110.0

-112.0

-114.0

-116.0

-118.0

-120.0

-122.0

-124.0

-126.0

-128.0

-130.0

-132.0

-134.0

-136.0

-138.0

-140.0

-142.0

-144.0

-146.0

-148.0

-150.0

-152.0

-154.0

-156.0

-158.0

-160.0

-162.0

-164.0

-166.0

-168.0

-170.0

-172.0

-174.0

-176.0

-178.0

-180.0

-182.0

-184.0

-186.0

-188.0

-190.0

-192.0

-194.0

-196.0

-198.0

-200.0

-202.0

-204.0

-206.0

-208.0

-210.0

-212.0

-214.0

-216.0

-218.0

-220.0

-222.0

-224.0

-226.0

-228.0

-230.0

-232.0

-234.0

-236.0

-238.0

-240.0

-242.0

-244.0

-246.0

-248.0

-250.0

-252.0

-254.0

-256.0

-258.0

-260.0

-262.0

-264.0

-266.0

-268.0

-270.0

-272.0

-274.0

-276.0

-278.0

-280.0

-282.0

-284.0

-286.0

-288.0

-290.0

-292.0

-294.0

-296.0

-298.0

-300.0

-302.0

-304.0

-306.0

-308.0

-310.0

-312.0

-314.0

-316.0

-318.0

-320.0

-322.0

-324.0

-326.0

-328.0

-330.0

-332.0

-334.0

-336.0

-338.0

-340.0

-342.0

-344.0

-346.0

-348.0

-350.0

-352.0

-354.0

-356.0

-358.0

-360.0

-362.0

-364.0

-366.0

-368.0

-370.0

-372.0

-374.0

-376.0

-378.0

-380.0

-382.0

-384.0

-386.0

-388.0

-390.0

-392.0

-394.0

-396.0

-398.0

-400.0

-402.0

-404.0

-406.0

-408.0

-410.0

-412.0

-414.0

-416.0

-418.0

-420.0

-422.0

-424.0

-426.0

-428.0

-430.0

-432.0

-434.0

-436.0

-438.0

-440.0

-442.0

-444.0

-446.0

-448.0

-450.0

-452.0

-454.0

-456.0

-458.0

-460.0

-462.0

-464.0

-466.0

-468.0

-470.0

-472.0

-474.0

-476.0

-478.0

-480.0

-482.0

-484.0

-486.0

-488.0

-490.0

-492.0

-494.0

-496.0

-498.0

-500.0

-502.0

-504.0

-506.0

-508.0

-510.0

-512.0

-514.0

-516.0

-518.0

-520.0

-522.0

-524.0

-526.0

-528.0

-530.0

-532.0

-534.0

-536.0

-538.0

-540.0

-542.0

-544.0

-546.0

-548.0

-550.0

-552.0

-554.0

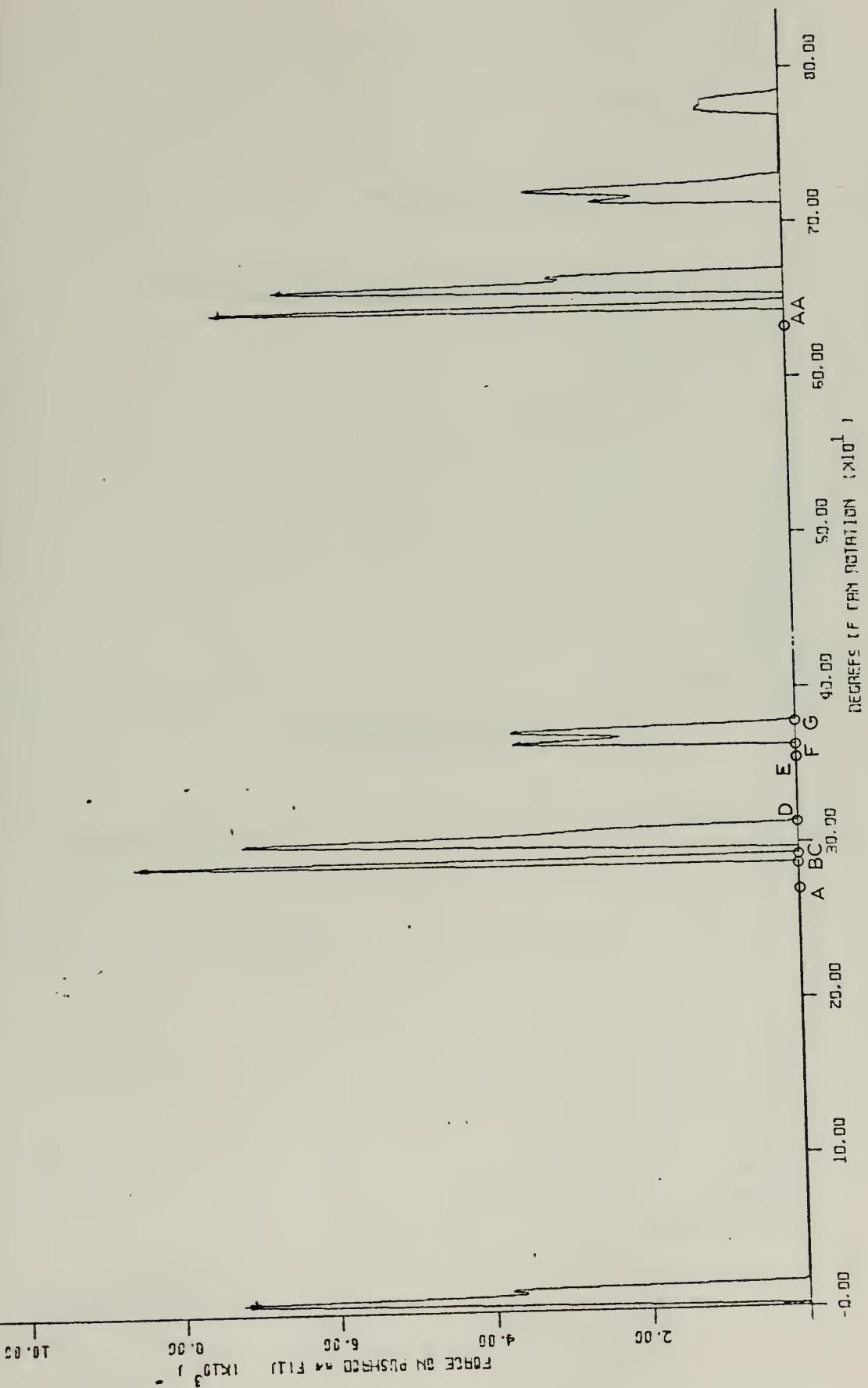


Figure 6.37 THREE CLEARANCES

11,000 RPM NO VALUE SEAT

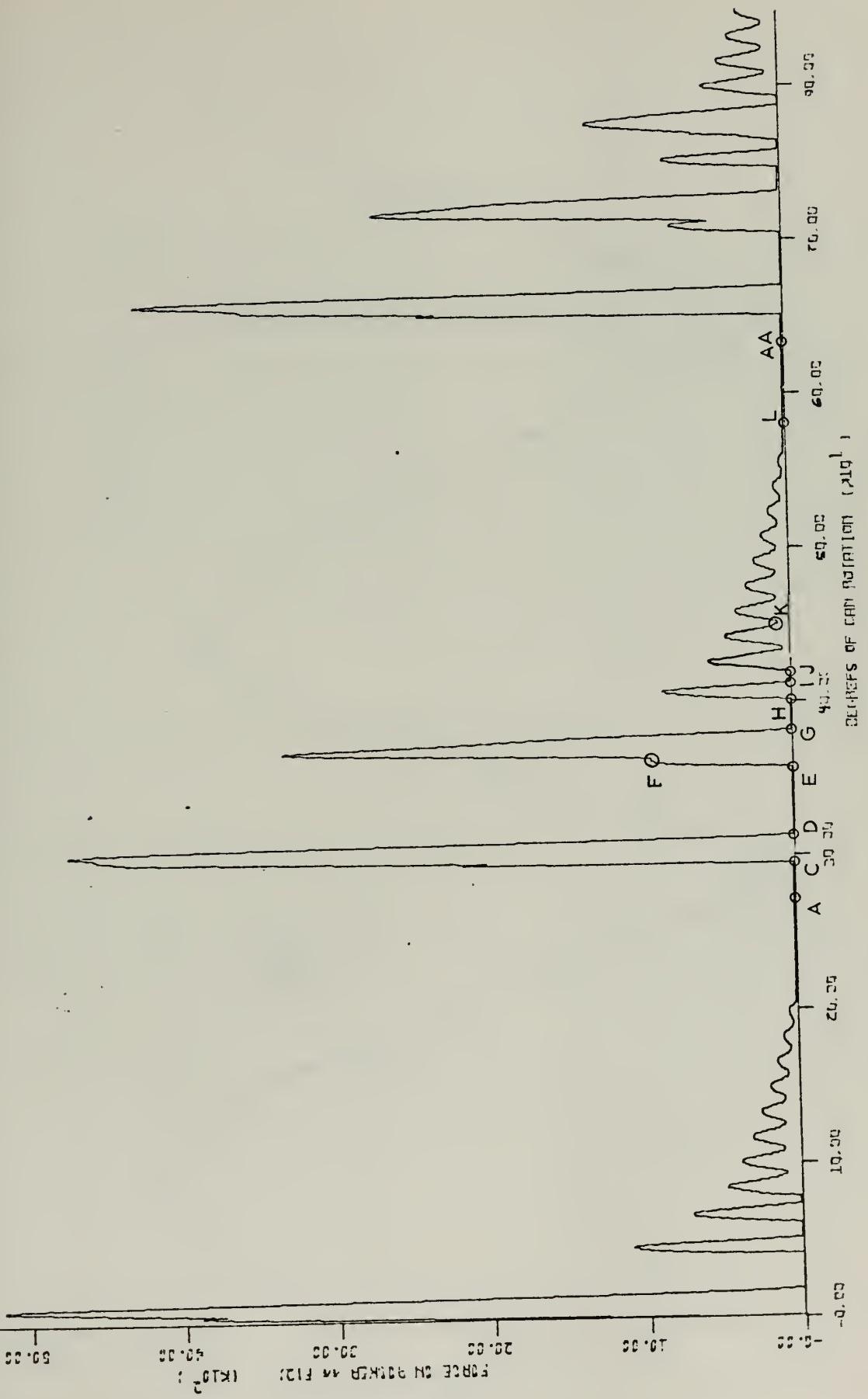


Figure 6.38 THREE CLEARANCES 11,000 RPM NO VALUE SEAT

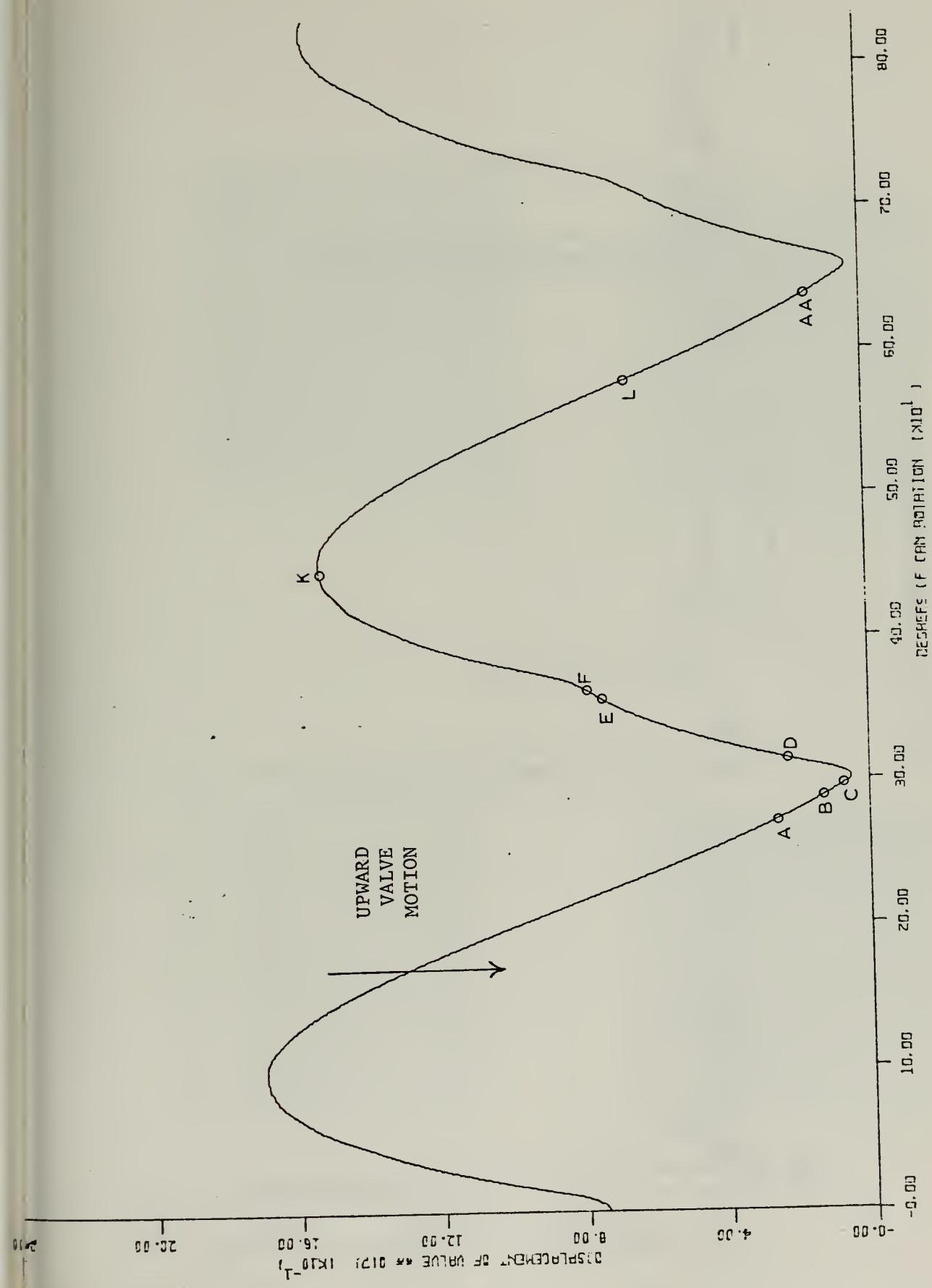


Figure 6.39 THREE CLEARANCES 11,000 RPM NO VALVE SEAT

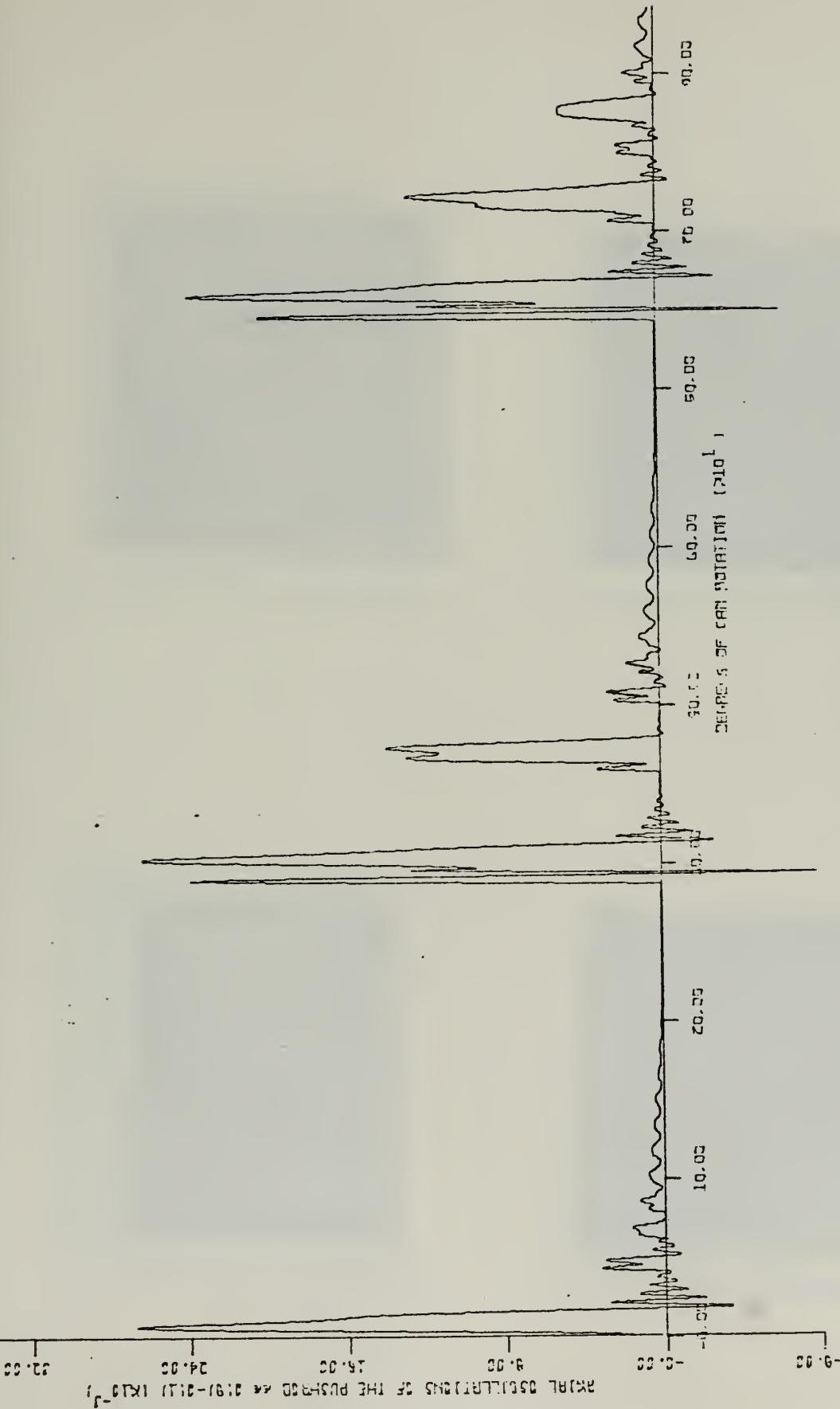


Figure 6.40 THREE CLEARANCES

11,000 RPM NO VALUE SEAT

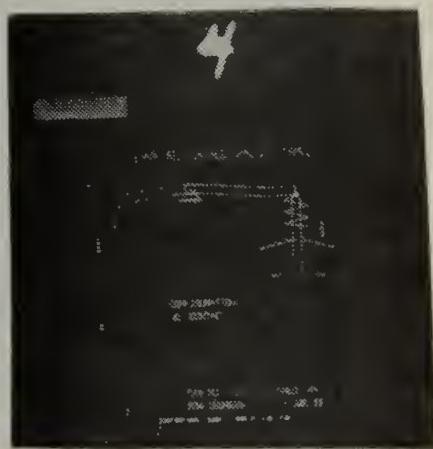


Photo 6.24 - 634 degrees



Photo 6.25 - 646 degrees



Photo 6.26 - 652 degrees



Photo 6.27 - 654 degrees



Photo 6.28 - 658 degrees

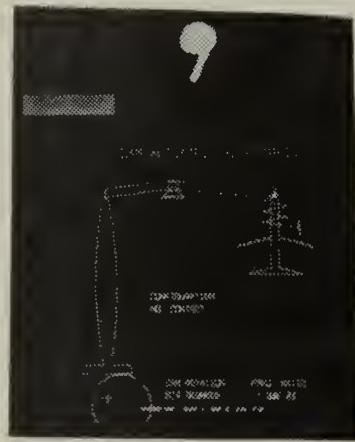


Photo 6.29 - 674 degrees



Photo 6.30 - 708 degrees



Photo 6.31 - 716 degrees

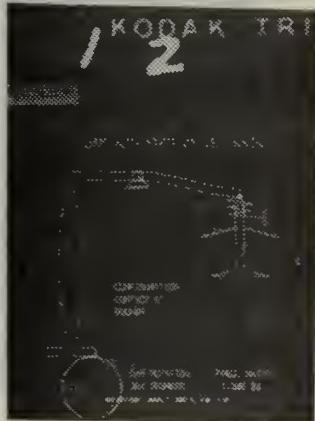


Photo 6.32 - 364 degrees

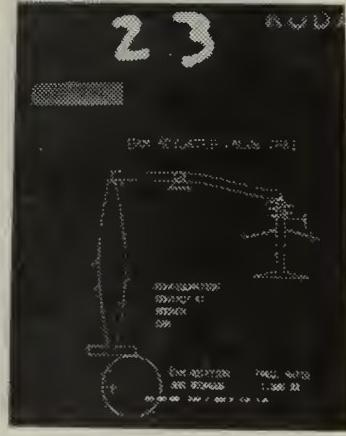


Photo 6.33 - 366 degrees

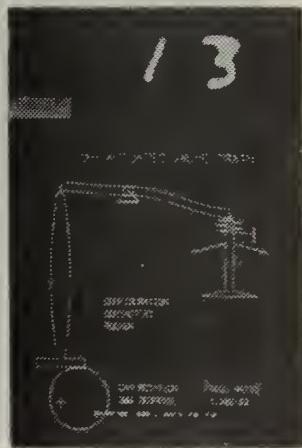


Photo 6.34 - 382 degrees

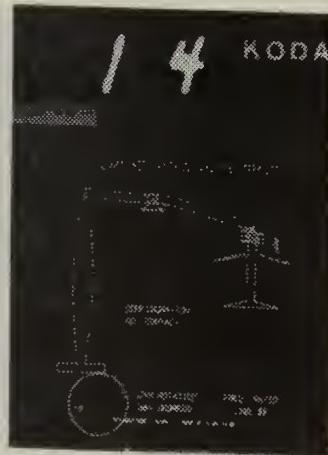


Photo 6.35 - 384 degrees

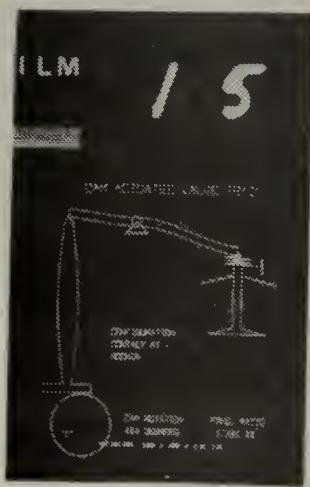


Photo 6.36 - 404 degrees

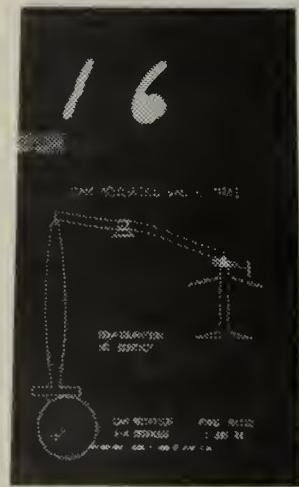


Photo 6.37 - 414 degrees



Photo 6.38 - 422 degrees



Photo 6.39 - 450 degrees

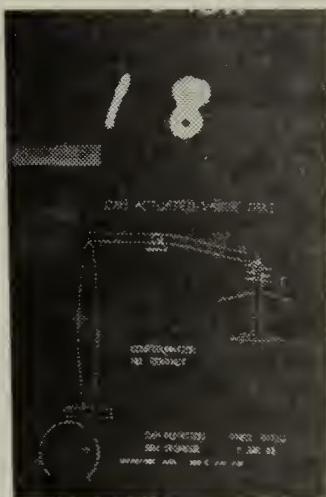


Photo 6.40 - 464 degrees



Photo 6.41 - 572 degrees

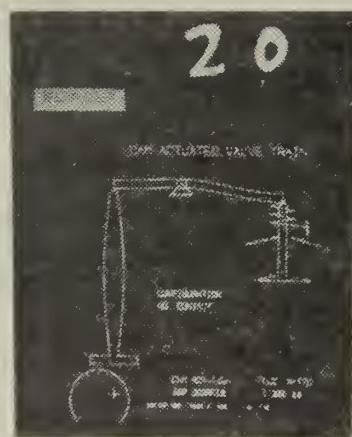


Photo 6.42 - 582 degrees

3. Example 1C, Two Clearances, 11,000 RPM, Valve Seat

This analysis will commence at 333 degrees of cam rotation. One cycle will be discussed, from point (A) to point (AA). The following figures are utilized for the analysis:

- a. Figure 6.41 Cam profile and displacement of pushrod,

$$q_9 \times 10^{-1} \text{ and } q_1 \times 10^{-1}$$

- b. Figure 6.42

(1) Cam profile, $q_9 \times 10^{-1}$

(2) Displacement difference between cam and pushrod,

$$(q_1 - q_9) \times 10^{-2}$$

- c. Figure 6.43 Force on pushrod, $F_1 \times 10^3$

- d. Figure 6.44 Force on the valve, $F_7 \times 10^3$

- e. Figure 6.45 Displacement of valve and valve seat,

$$q_7 \times 10^{-1} \text{ and } q_{10} \times 10^{-1}$$

- f. Figure 6.46 Axial oscillation of the pushrod,

$$(q_8 - q_1) \times 10^{-2}$$

- g. Figure 6.47 Axial oscillation of the valve stem,

$$(q_7 - q_6) \times 10^{-3}$$

Figures 6.46 and 6.47 have been included for completeness and are not annotated.

The following reference points approximately correspond to the indicated degrees of cam rotation, Table 6.3.

Reference Point	Degrees of Cam Rotation	Reference Point	Degrees of Cam Rotation
A	333	H	600
B	340	I	606
C	379	J	612
D	398	K	681
E	448	L	688
F	467	AA	693
G	483		

Table 6.3 Graph Reference Points and Degrees of Cam Rotation
For Figures 6.41 - 6.45

The analysis is similar to that for examples 1A and 1B. For this example the pushrod is pinned to the rocker and thus moves as a part of the system rather than floating free.

At point (A), impact against the pushrod by the cam has just occurred. This has created large system oscillations, first in the pushrod and later in the valve stem. The cam and pushrod are rising. At points (B) and (D) the pushrod is thrown clear of the cam due to its high speed of rotation. At point (E) the cam has reached its highest position and starts to fall. The valve has reached its lowest position and starts its upward travel.

At point (F) the pushrod and rocker come into contact with very little force and travel together for a short distance. Separation again occurs at point (G) when the cam falls away from the pushrod. The valve in its upward travel impacts against the valve seat at point (H). This retards the motion of the pushrod just before contact

at the cam was due to occur (I). The large oscillations created by the valve seating cause the valve to bounce and leave contact at the seat, point (J).

Meanwhile the cam has reached its lowest extreme and begins its upward travel. At point (K) the rising cam impacts against the oscillating pushrod and forces the system to return to its harmonic motion. The cycle then repeats, point (AA).

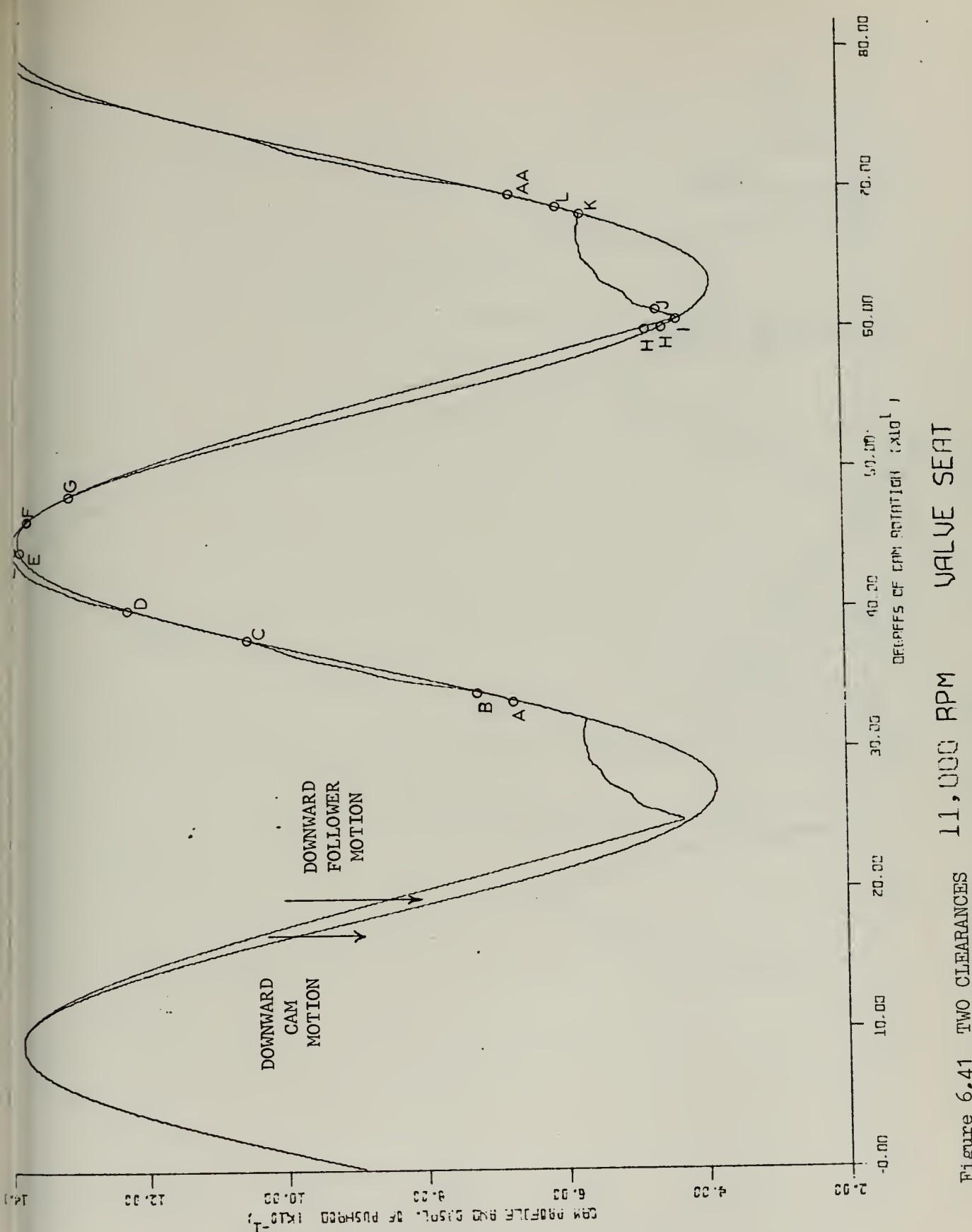


Figure 6.41 TWO CLEARANCES 11,000 RPM

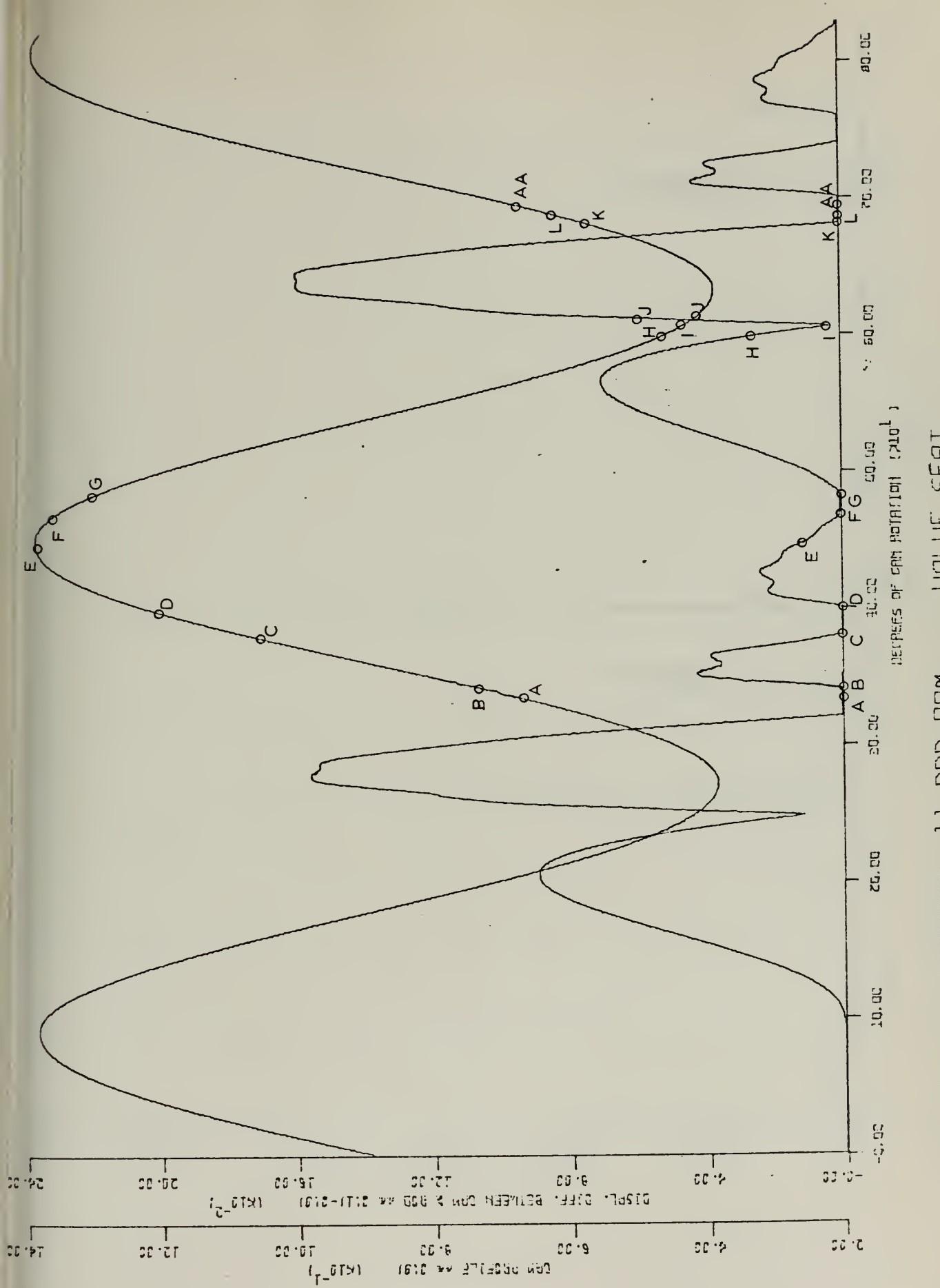


Figure 6.42 TWO CLEARANCES 11,000 RPM VALUE SEH

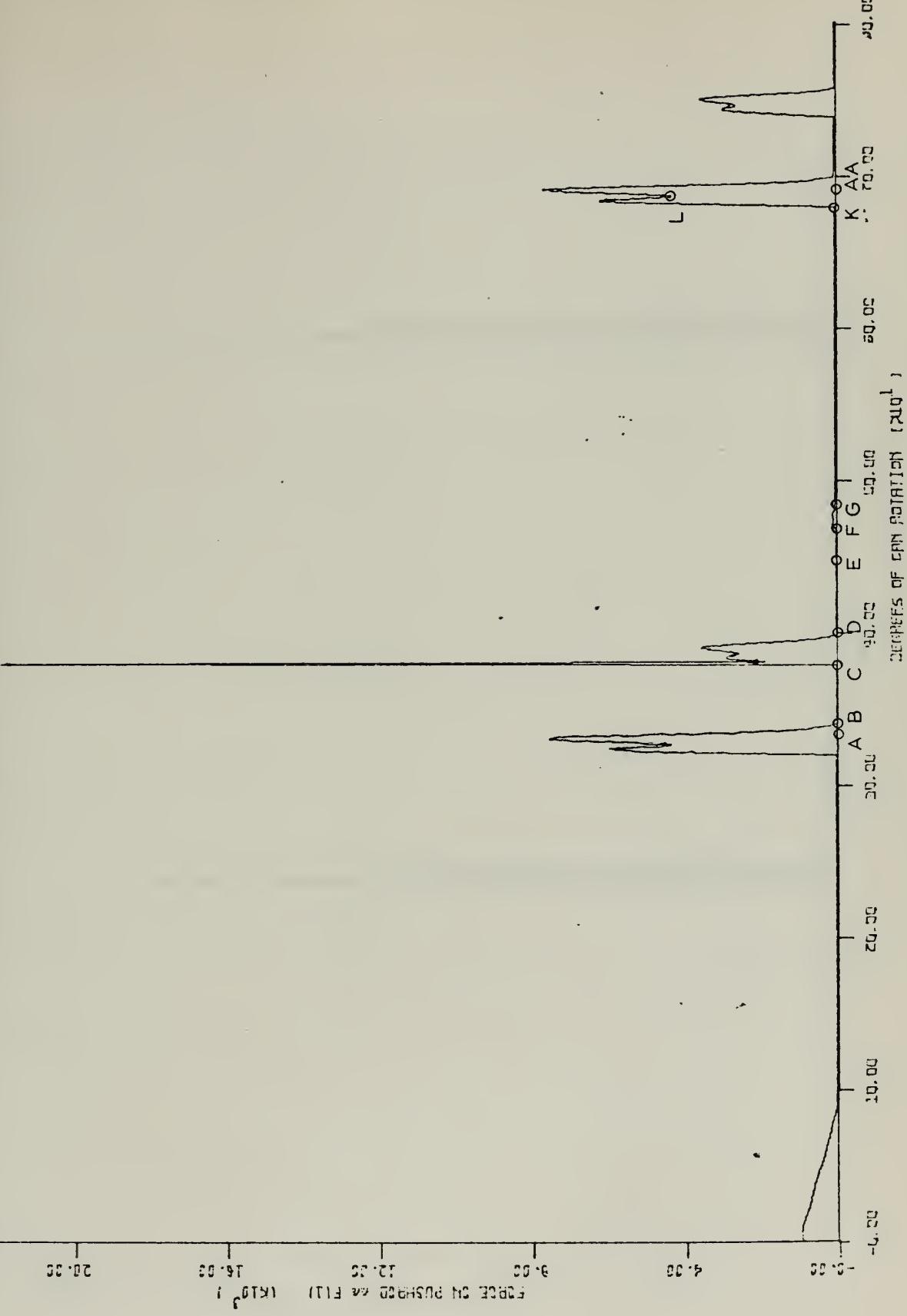


Figure 6.43 TWO CLEARANCES

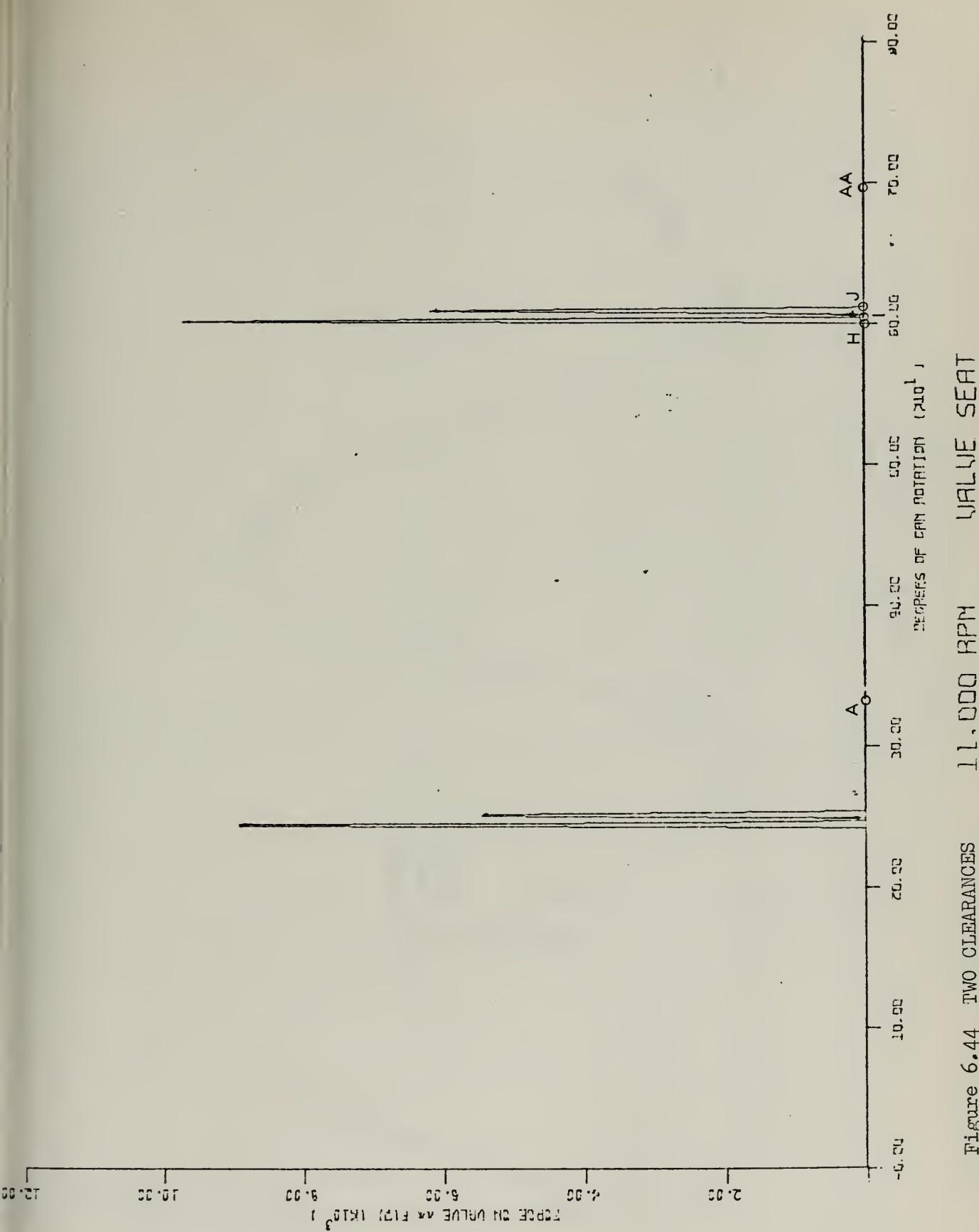


Figure 6.44 TWO CLEARANCES

11,000 RPM VALUE SEAT

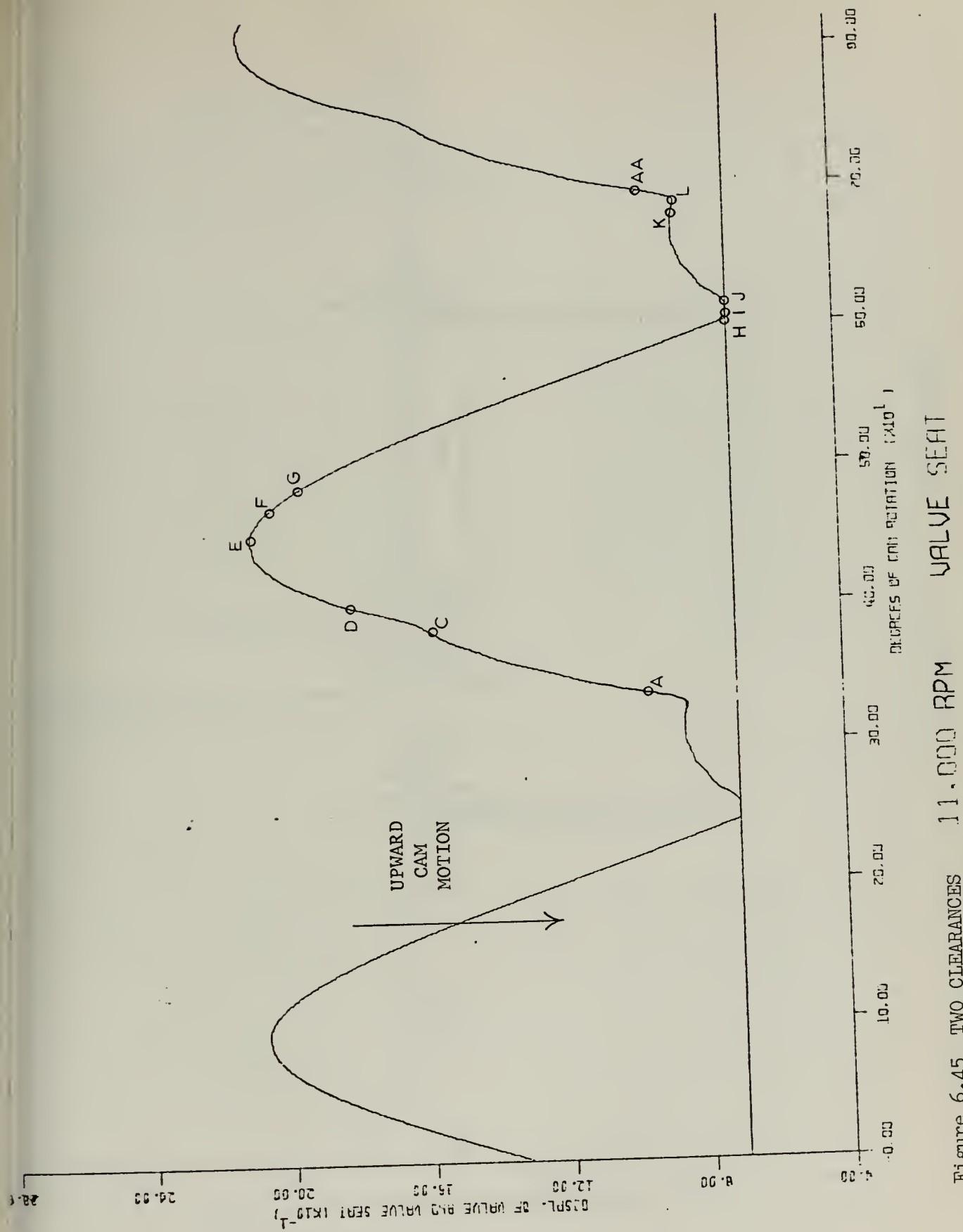
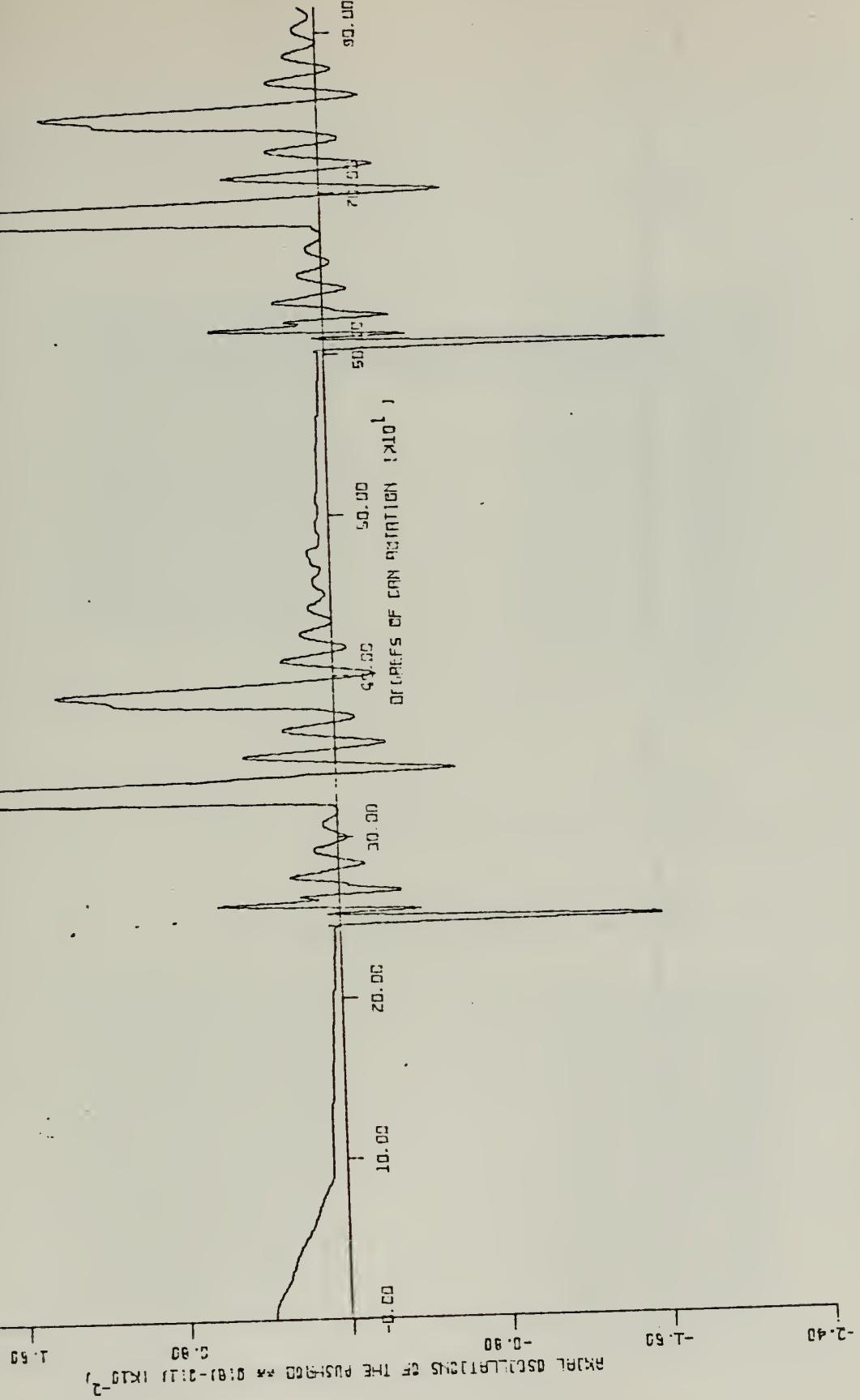


Figure 6.45 TWO CLEARANCES

VALVE SEAT

11,000 RPM

TWO CLEARANCES



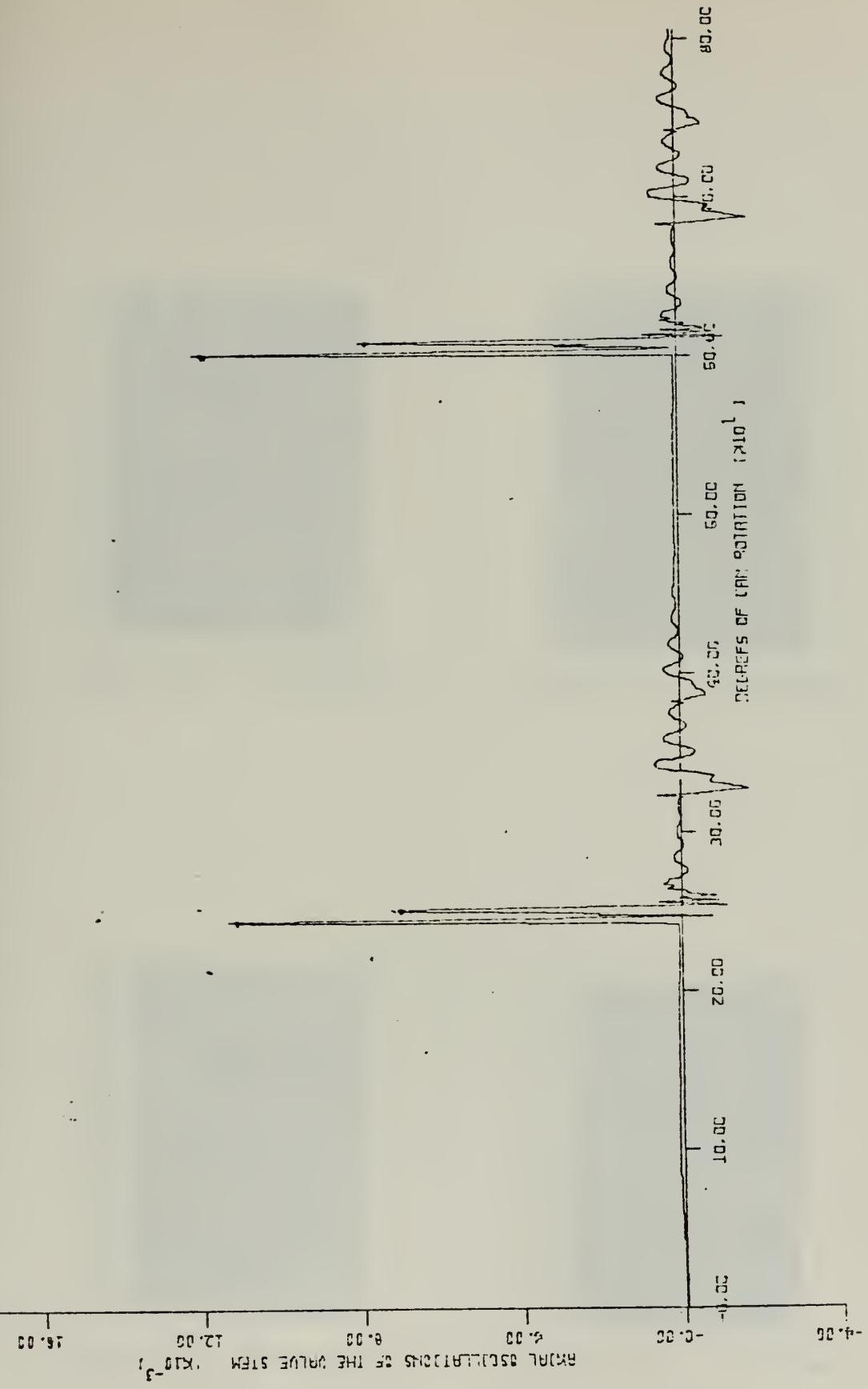


Figure 6.47 TWO CLEARANCES

11,000 RPM VALUE STEM



Photo 6.43 - 700 degrees



Photo 6.44 - 702 degrees

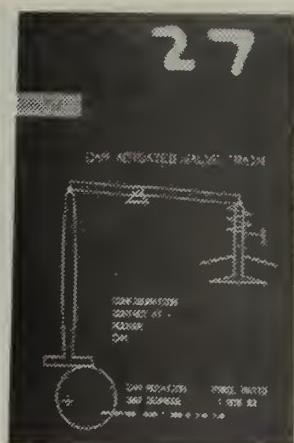


Photo 6.45 - 382 degrees



Photo 6.46 - 400 degrees

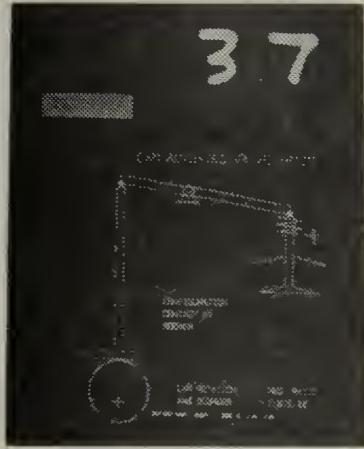


Photo 6.47 - 448 degrees

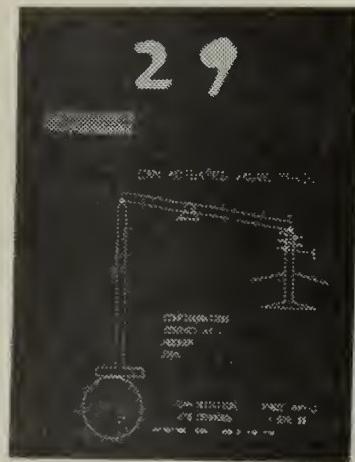


Photo 6.48 - 470 degrees

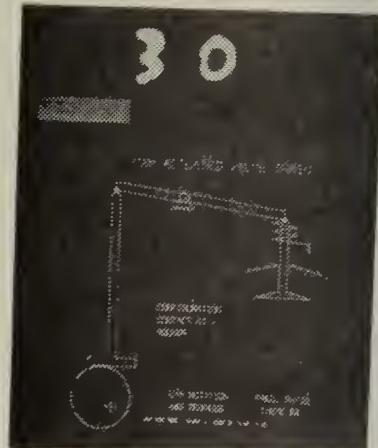


Photo 6.49 - 482 degrees

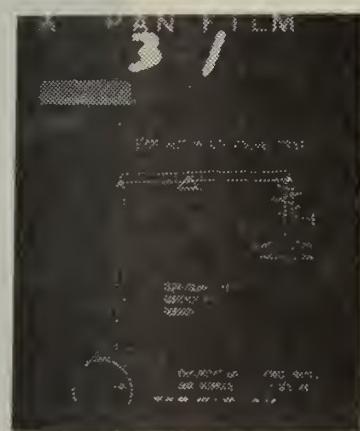


Photo 6.50 - 560 degrees

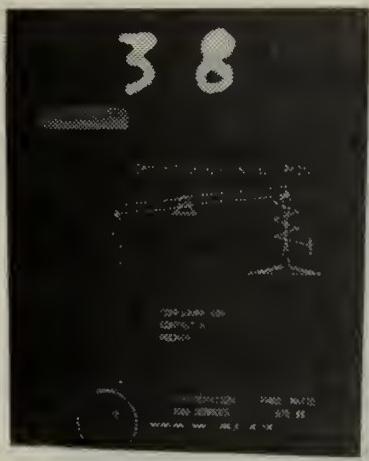


Photo 6.51 - 602 degrees

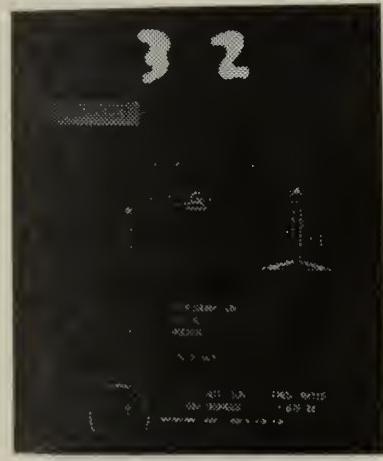


Photo 6.52 - 604 degrees



Photo 6.53 - 606 degrees

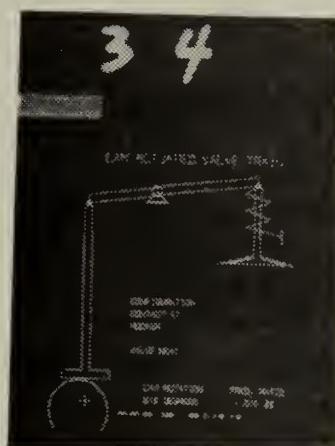


Photo 6.54 - 610 degrees

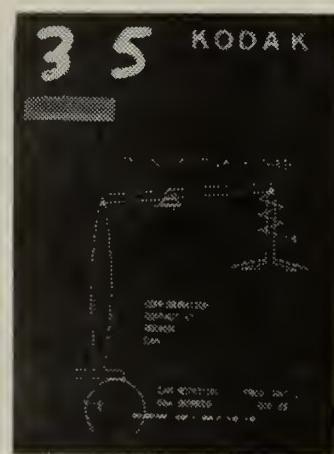


Photo 6.55 - 684 degrees

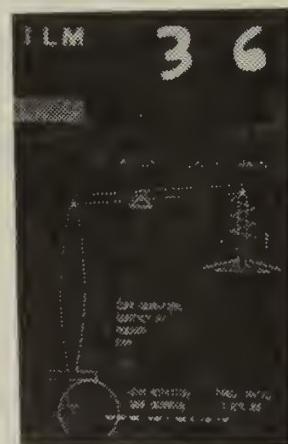


Photo 6.56 - 690 degrees

4. Example 1D, Two Clearances, 11,000 RPM, No Valve Seat

This analysis will commence at 274 degrees of cam rotation, point

(A). The following figures are utilized for the analysis.

a. Figure 6.48

(1) Cam profile, $q_9 \times 10^{-1}$

(2) Displacement of lower end of pushrod,

$q_1 \times 10^{-1}$

(3) Displacement difference between cam and pushrod,

$(q_1 - q_9) \times 10^{-2}$

b. Figure 6.49, Force on pushrod, $F_1 \times 10^3$

c. Figure 6.50, Displacement of valve, $q_7 \times 10^{-1}$

d. Figure 6.51, Axial oscillation of the pushrod,

$(q_9 - q_1) \times 10^{-2}$

Figure 6.51 has been included for completeness and is not annotated.

The following reference points approximately correspond to the indicated degrees of cam rotation, Table 6.4.

Reference Point	Degrees of Cam Rotation
A	274
B	295
C	314
D	376

Reference Point	Degrees of Cam Rotation
E	398
F	448
AA	634

Table 6.4 Graph Reference Points and Degrees of Cam Rotation For Figures 6.48 - 6.50

The analysis is similar to the above examples. The valve seat position is higher than the maximum excursion of the valve, thus contact does not occur between the seat and valve. The remaining points of interest are then only associated with clearance l , between pushrod and cam.

At point (A) the cam has reached its lowest extreme and begins rising. The pushrod continues its downward harmonic motion, the valve its harmonic motion upward. At point (B), the rising cam impacts against the pushrod. The motion of the pushrod is reversed. The severe oscillations and inertial effects first insure positive contact and then cause the pushrod and cam to separate, point (C). As the upward motion of the pushrod slows, the rising cam again contacts the pushrod, point (D). Since the motion of these two members is similar, the impact force is not as severe.

At point (E), the speed of cam rotation throws the pushrod off. Harmonic motion of the system then results. The cycle repeats at point (AA).

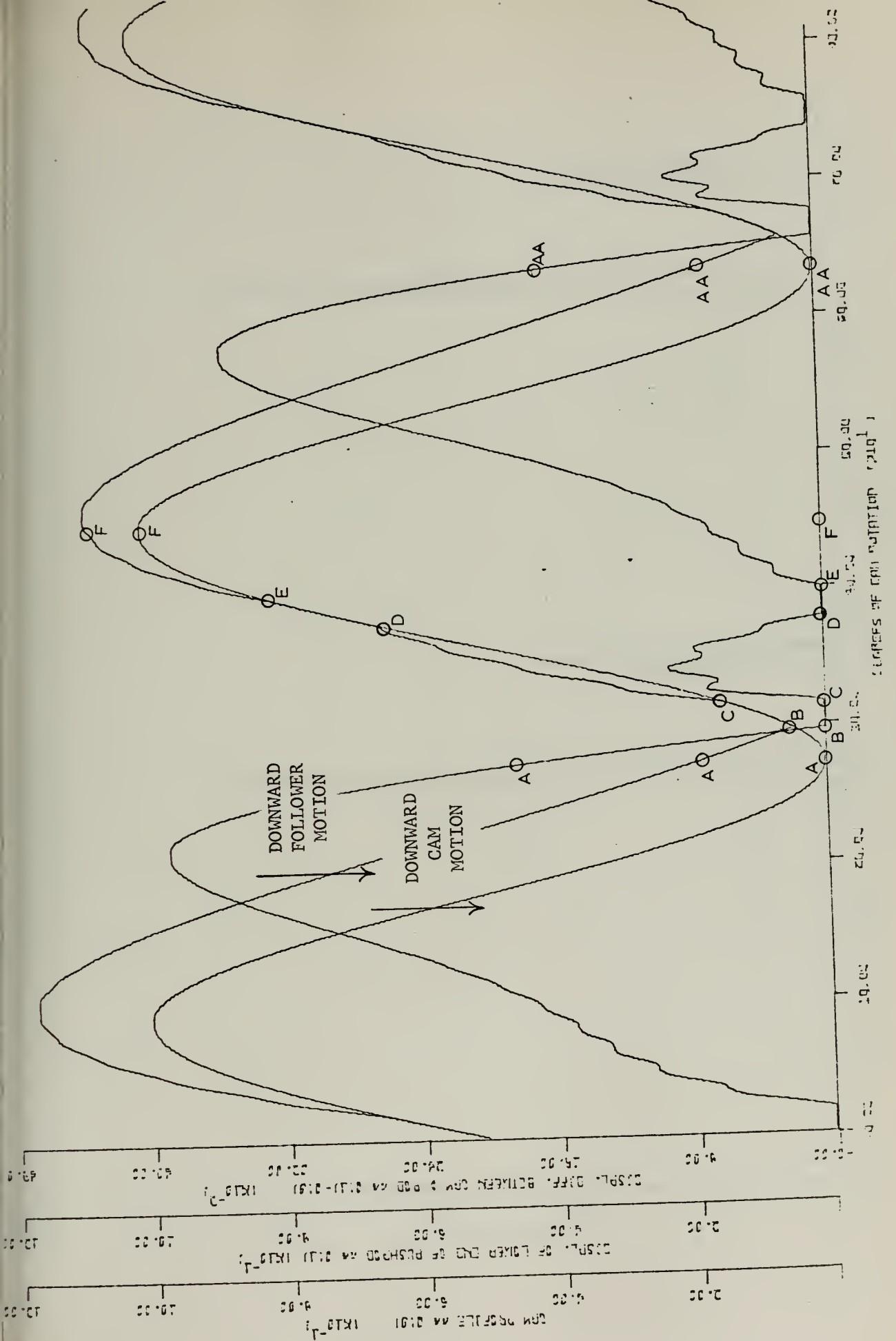


Figure 6.48 TWO CLEARANCES

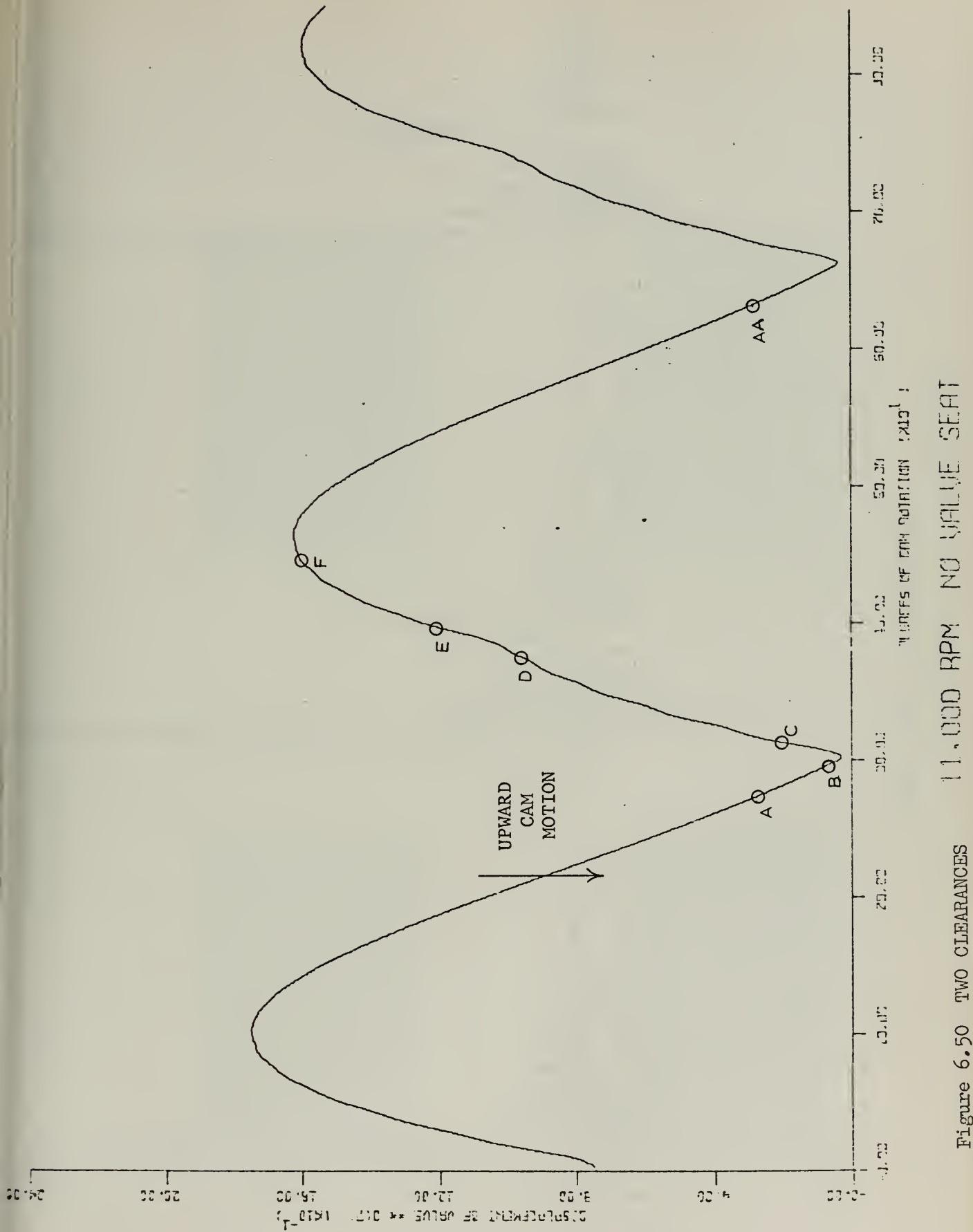


Figure 6.50 TWO CLEARANCES

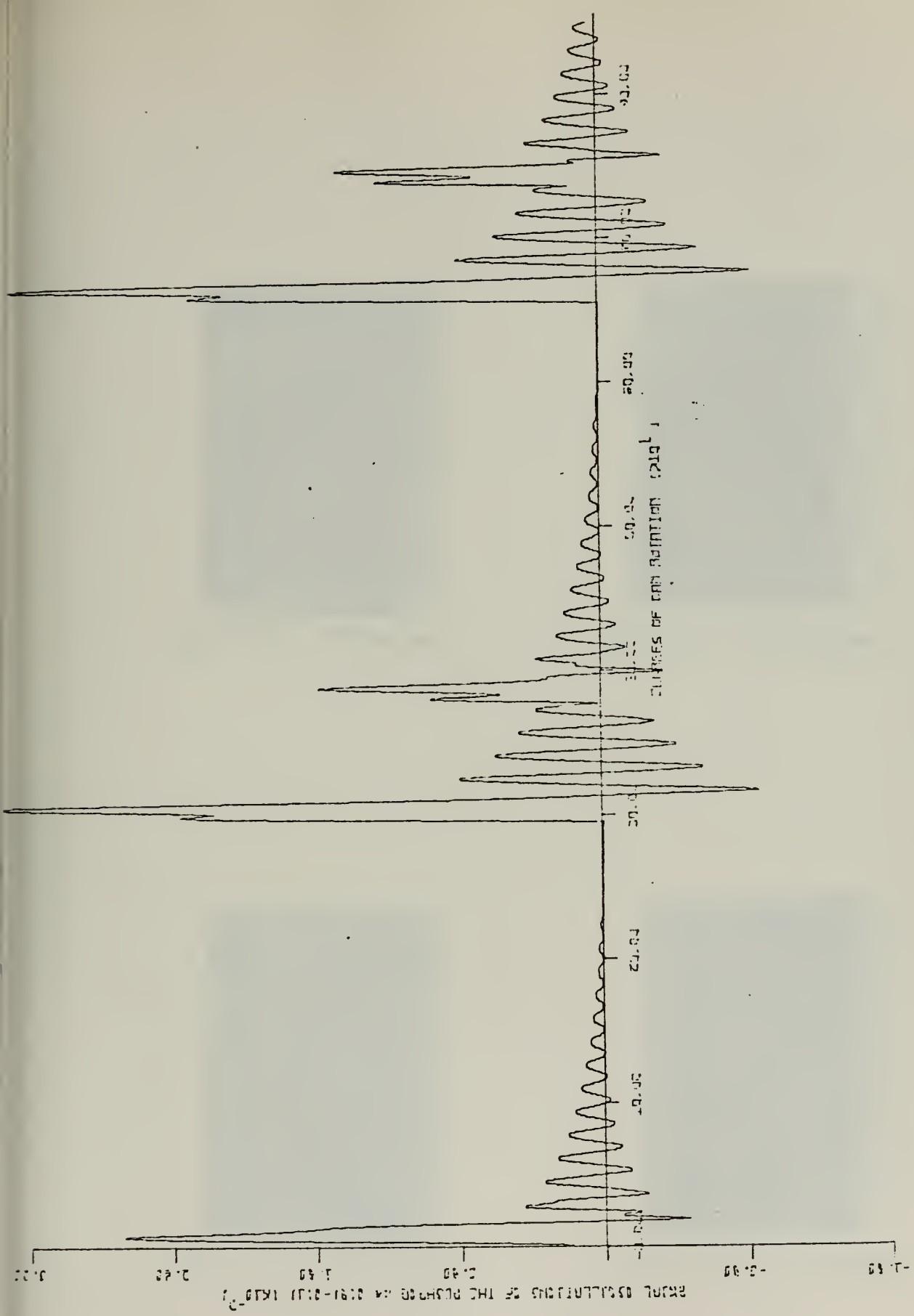


Figure 6.51 TWO CLEARANCES
11,000 RPM NO VALUE SEAT

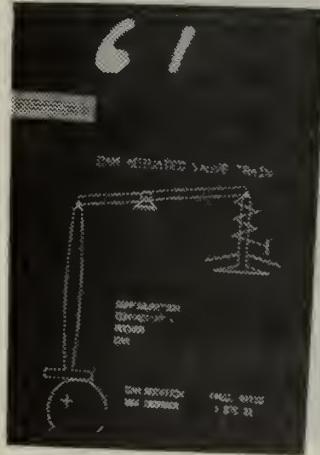


Photo 6.57 - 664 degrees

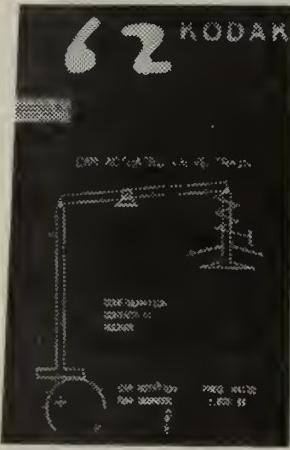


Photo 6.58 - 682 degrees



Photo 6.59 - 694 degrees

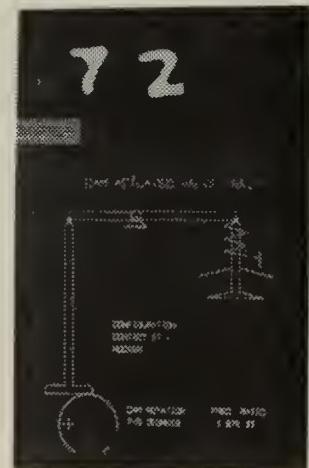


Photo 6.60 - 716 degrees

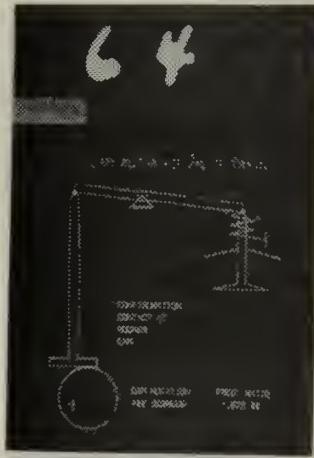


Photo 6.61 - 400 degrees

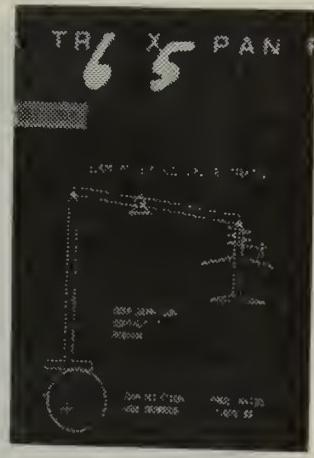


Photo 6.62 - 420 degrees

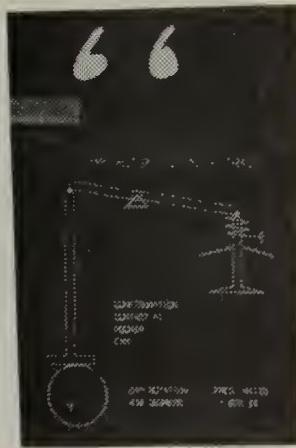


Photo 6.63 - 430 degrees



Photo 6.64 - 452 degrees



Photo 6.65 - 466 degrees

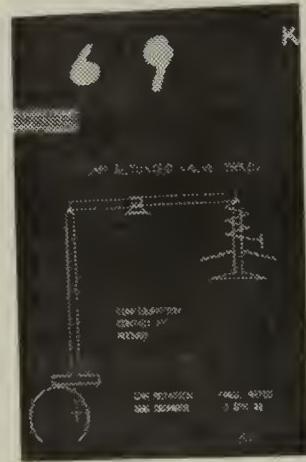


Photo 6.67 - 586 degrees

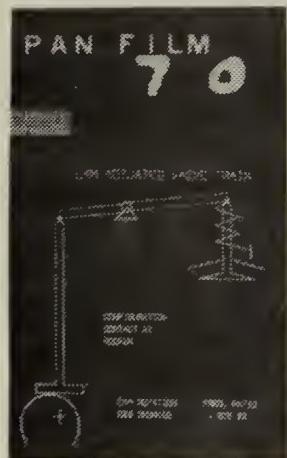


Photo 6.66 - 600 degrees

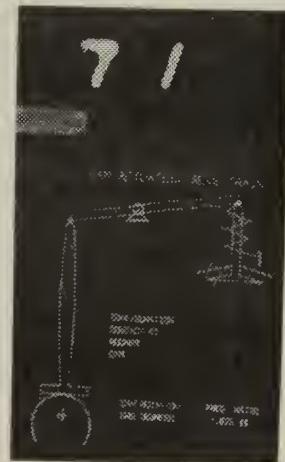


Photo 6.68 - 626 degrees

5. Example two, 9000 RPM and example three, 10,000 RPM were chosen for comparison and completeness of this study. These examples are not annotated and may be found in Appendix F.

VII. CONCLUSION

The simulation for the dynamic response of an elastic link model with three clearances has been described. The algorithm developed is applied to the automobile cam actuated valve train assembly, with external clearances at the valve and cam. The internal clearance at the pushrod-left rocker arm, requires the solution to a second eigenvalue problem in a portion of the calculations. These algorithms are generally coded to allow for application to similar mechanisms. The methods of calculation use address modification arrays to enable application of the program to more complicated models by changing a minimum number of variables and array elements. The numerous subroutines developed have, in general, very specific functions, and are intended to be used as separate calculation packages in a variety of other applications.

While the accuracy of the mathematical model of a mechanism has been greatly increased over the rigid version, the complexity of solution has also increased, even though a number of simplifying assumptions were made. The major assumption made by the theory was that of a linear model so as to permit superposition. Other assumptions, made to simplify the equations but not the theory, involved the choice of coordinates and the use of straight, constant area links.

The purpose of this work was to demonstrate a more accurate model of a mechanism. The extent of this work was to (1) restructure and reprogram an existing simulation to provide for a generalized

method of establishing the physical model configurations; (2) Modify this simulation to allow for more than two clearances; (3) Establish a method for the dynamic graphical display of the model showing the resulting elastic deformations of the model members at high operational speeds. Objectives (1) and (2) are accomplished using the IBM-360 computer for a three clearance simulation. Dynamic dimensioning and program address modifications arrays contribute to the program generalization. Sample problem graphical output is provided. Objective (3) is accomplished with the aid of the ADAGE Graphical Terminal, model 10 interfaced with a Scientific Data System Computer model 9300. Data generated by the IBM computer is transferred to the SDS computer by punched cards. A program then calculates the model's geometry, position, and deformations. The results are then displayed on the AGT. These results are provided on video tape and may be viewed by requesting NPGS Video Tape 12NPS7200042-VR from the Educational Media Department, Naval Postgraduate School.

It is suggested that additional study be done on the calculation of the pushrod bounce at the rocker. The simulation, as written, has difficulty in obtaining a stable system when contact is regained at this point.

An extension of the AGT system would be to allow for the display of force and displacement curves on the AGT. Methods similar to those developed by this author and Lt. Bowden (Ref. 19) could be conveniently applied.

APPENDIX A

MASS AND STIFFNESS MATRICES

The system mass and stiffness matrices are given by equations (3.6) and (3.7), and are repeated below.

$$[M] = \sum_{i=1}^5 [\beta]_i^T [M_e]_i [\beta]_i \quad (3.6)$$

$$[K] = \sum_{i=1}^5 [\beta]_i^T [K_e]_i [\beta]_i \quad (3.7)$$

The transformantion matrix $[\beta]$ is given in equation (3.4) and will not be repeated here. The element stiffness and mass matrices are given by reference (1). For the axial members (pushrod, spring, and valve) they are:

$$[K_e]_i = \frac{E_i A_i}{L_i} \quad \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad (A.1)$$

$$[M_e]_i = \frac{M_i}{6} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \quad (i=1,4,5) \quad (A.2)$$

The element stiffness and mass matrices for the bending members (left and right arms) are:

$$[K_e]_i = \frac{E_i I_i}{L_i^3} \quad \begin{bmatrix} 12 & -12 & 6L_i & 6L_i \\ -12 & 12 & -6L_i & -6L_i \\ 6L_i & -6L_i & 4L_i^2 & 2L_i^2 \\ 6L_i & -6L_i & 2L_i^2 & 4L_i^2 \end{bmatrix} \quad (A.3)$$

$$[M_e]_i = \frac{M_i}{420}$$

156	54	$22L_i$	$-13L_i$	(A.4)
54	156	$13L_i$	$-22L_i$	(i=2,3)
$22L_i$	$13L_i$	$4L_i^2$	$-3L_i^2$	
$-13L_i$	$-22L_i$	$-3L_i^2$	$4L_i^2$	

For equations A.1 through A.4, E_i is the element elastic modulus, A_i is the element cross-sectional area, L_i is the element length, I_i is the element moment of inertia about the bending axis, and M_i is the total element mass of the i^{th} member.

Evaluating equation (3.6) for the five elements of the system yields the full system mass matrix, designated as layer one. The elements of this matrix are given by Figure A.1. Similarly, evaluating equation (3.7) yields the system stiffness matrix. The elements of this matrix are shown in Figure A.2.

$\frac{M_1}{3}$	0	0	0	0	0	0	$\frac{M_1}{6}$
0	$\frac{156 M_2}{420}$	$\frac{22 M_2 L_2}{420}$	0	0	$\frac{-13 L_2 M_2}{420}$	0	0
0	$\frac{22 L_2 M_2}{420}$	$\frac{4 L_2^2 M_2}{420}$	0	0	$\frac{-3 L_2^2 M_2}{420}$	0	0
0	0	0	$\frac{156 M_3}{420} + \frac{M_4}{3}$	$\frac{-22 L_3 M_3}{420}$	$\frac{13 L_3 M_3}{420}$	$\frac{-M_5}{6}$	0
		$\frac{+M_5}{3}$					
0	0	0	$\frac{-22 L_3 M_3}{420}$	$\frac{4 L_3^2 M_3}{420}$	$\frac{-3 L_3^2 M_3}{420}$	0	0
0	$\frac{-13 L_2 M_2}{420}$	$\frac{-3 L_2^2 M_2}{420}$	$\frac{13 L_3 M_3}{420}$	$\frac{-3 L_3^2 M_3}{420}$	$\frac{4 L_2^2 M_2}{420}$	0	0
0	0	0	$\frac{-M_5}{6}$	0	0	$\frac{M_5}{3}$	0
$\frac{M_1}{6}$	0	0	0	0	0	0	$\frac{M_1}{3}$

Figure A.1 Full System Mass Matrix

$$\begin{matrix}
\frac{E_1 A_1}{L_1} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-E_1 A_1}{L_1} \\
0 & \frac{+12E_2 I_2}{L_2^3} & \frac{6E_2 I_2}{L_2^2} & 0 & 0 & \frac{6E_2 I_2}{L_2^2} & 0 & 0 \\
0 & \frac{6E_2 I_2}{L_2^2} & \frac{4E_2 I_2}{L^2} & 0 & 0 & \frac{2E_2 I_2}{L_2} & 0 & 0 \\
0 & 0 & 0 & \frac{12E_3 I_3}{L_3^3} & \frac{-6E_3 I_3}{L_3^2} & \frac{-6E_3 I_3}{L_3^2} & \frac{E_5 A_5}{L_5} & 0 \\
& & & \frac{+E_5 A_5}{L_5} & & & & \\
& & & \frac{+E_4 A_4}{L_4} & & & & \\
0 & 0 & 0 & \frac{-6E_3 I_3}{L_3^2} & \frac{4E_3 I_3}{L_3} & \frac{2E_3 I_3}{L_3} & 0 & 0 \\
0 & \frac{6E_2 I_2}{L_2^2} & \frac{2E_2 I_2}{L_2} & \frac{-6E_3 I_3}{L_3^2} & \frac{2E_3 I_3}{L_3} & \frac{4E_3 I_3}{L_3} & 0 & 0 \\
& & & & & \frac{+4E_2 I_2}{L_2} & & \\
0 & 0 & 0 & \frac{E_5 A_5}{L_5} & 0 & 0 & \frac{E_5 A_5}{L_5} & 0 \\
\frac{-E_1 A_1}{L_1} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{E_1 A_1}{L_1}
\end{matrix}$$

Figure A.2 Full System Stiffness Matrix

The mass and stiffness matrices are valid for the eight coordinate system, i.e. configuration I, II, V, and VI. When the rocker is in contact with the pushrod, the seven coordinate system is used and the transformation, mass and stiffness revert to those developed by Anderson, reference (2). The reduced system mass and stiffness matrices are shown in Figures A.3 and A.4, respectfully. These reduced matrices are stored in layer 3 and are used for configurations III, IV, VII, and VIII.

$\frac{M_1}{3}$	$\frac{-M_1}{6}$	0	0	0	0	0
$\frac{-M_1}{6}$	$\frac{M_1}{3} + \frac{156M_2}{420}$	$\frac{22M_2L_2}{420}$	0	0	$\frac{-13L_2M_2}{420}$	0
0	$\frac{22L_2M_2}{420}$	$\frac{4L_2^2M_2}{420}$	0	0	$\frac{-3L_2^2M_2}{420}$	0
0	0	0	$\frac{156M_3}{420} + \frac{M_4}{3}$	$\frac{-22L_3M_3}{420}$	$\frac{13L_3M_3}{420}$	$\frac{-M_5}{6}$
			$\frac{+M_5}{3}$			
0	0	0	$\frac{-22L_3M_3}{420}$	$\frac{4L_3^2M_3}{420}$	$\frac{-3L_3^2M_3}{420}$	0
0	$\frac{-13L_2M_2}{420}$	$\frac{-3L_2^2M_2}{420}$	$\frac{13L_3M_3}{420}$	$\frac{-3L_3^2M_3}{420}$	$\frac{4L_2^2M_2}{420}$	0
0	0	0	$\frac{-M_5}{6}$	0	0	$\frac{M_5}{3}$

Figure A.3 Reduced System Mass Matrix

$\frac{E_1 A_1}{L_1}$	$\frac{E_1 A_1}{L_1}$	0	0	0	0	0
$\frac{E_1 A_1}{L_1}$	$\frac{E_1 A_1}{L_1}$	$\frac{6E_2 I_2}{L_2^2}$	0	0	$\frac{6E_2 I_2}{L_2^2}$	0
		$\frac{+12E_2 I_2}{L_2^3}$				
0	$\frac{6E_2 I_2}{L_2^2}$	$\frac{4E_2 I_2}{L_2}$	0	0	$\frac{2E_2 I_2}{L_2}$	0
0	0	0	$\frac{12E_3 I_3}{L_3^3}$	$\frac{-6E_3 I_3}{L_3^2}$	$\frac{-6E_3 I_3}{L_3^2}$	$\frac{E_5 A_5}{L_5}$
			$\frac{+E_5 A_5}{L_5}$			
			$\frac{+E_4 A_4}{L_4}$			
0	0	0	$\frac{-6E_3 I_3}{L_3^2}$	$\frac{4E_3 I_3}{L_3}$	$\frac{2E_3 I_3}{L_3}$	0
0	$\frac{6E_2 I_2}{L_2^2}$	$\frac{2E_2 I_2}{L_2}$	$\frac{-6E_3 I_3}{L_3^2}$	$\frac{2E_3 I_3}{L_3}$	$\frac{4E_3 I_3}{L_3}$	0
			$\frac{+4E_2 I_2}{L_2}$			
0	0	0	$\frac{E_5 A_5}{L_5}$	0	0	$\frac{E_5 A_5}{L_5}$

Figure A.4 Reduced System Stiffness Matrix

APPENDIX B

NEWMARK'S β PARAMETER METHOD

The one step algorithm developed by Newmark (Ref. 4), is chosen as the numerical integration technique used in the simulation. The β parameter may be changed for an analysis. This time marching technique solves the governing equations of motion by using the following relations:

$$\{\ddot{q}\}^{n+1} = \{\dot{q}\}^n + \frac{\Delta T}{2} (\{\ddot{q}\}^n + \{\ddot{q}\}^{n+1}) \quad (B.1)$$

and

$$\begin{aligned} \{q\}^{n+1} = & \{q\}^n + \Delta T \{\dot{q}\}^n + \frac{(\Delta T)^2}{2} \{\ddot{q}\}^n \\ & + \beta (\Delta T)^2 (\{\ddot{q}\}^{n+1} - \{q\}^n) \end{aligned} \quad | \quad (B.2)$$

The net effect of the parameter β is to change the form of the variation of acceleration during the time interval ΔT . For this study, $\beta = 1/6$, assumes that the acceleration changes linearly from $\{\ddot{q}\}^n$ to $\{\ddot{q}\}^{n+1}$, Figure B.1.

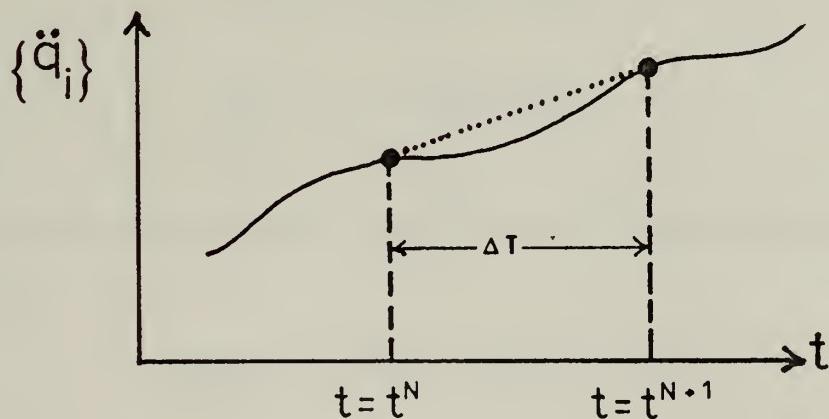


Figure B.1 Time Step Acceleration Curve

In this regard equations B.1 and B.2 are written as:

$$\text{Acceleration} = \ddot{q}_i(t) = \ddot{q}_i^n + \frac{(\dot{q}_i^{n+1} - \dot{q}_i^n)}{\Delta T} t \quad (\text{B.3})$$

$$\text{Velocity} = \dot{q}_i(t) = \dot{q}_i^n + \ddot{q}_i^n t + \frac{(\dot{q}_i^{n+1} - \dot{q}_i^n)}{\Delta T} \frac{t^2}{2} \quad (\text{B.4})$$

$$\text{Displacement} = q_i(t) = q_i^n + \dot{q}_i^n t + \ddot{q}_i^n \frac{t^2}{2} + \frac{(\dot{q}_i^{n+1} - \dot{q}_i^n)}{\Delta T} \frac{t^3}{6} \quad (\text{B.5})$$

Evaluate $\dot{q}_i(t)$ and $q_i(t)$ at time $t = \Delta T$,

$$\dot{q}_i^{n+1} = \dot{q}_i^n + \ddot{q}_i^n \frac{\Delta T}{2} + \ddot{q}_i^{n+1} \frac{\Delta T}{2} \quad (\text{B.6})$$

$$q_i^{n+1} = q_i^n + \dot{q}_i^n \Delta T + \ddot{q}_i^n \frac{(\Delta T)^2}{3} + \ddot{q}_i^{n+1} \frac{(\Delta T)^2}{6} \quad (\text{B.7})$$

$$\text{Let } \alpha_i^n = \dot{q}_i^n + \ddot{q}_i^n \frac{\Delta T}{2} \quad (\text{B.8})$$

$$\text{and } \psi_i^n = q_i^n + \dot{q}_i^n \Delta T + \ddot{q}_i^n \frac{(\Delta T)^2}{3} \quad (\text{B.9})$$

Now \dot{q}_i^{n+1} and q_i^{n+1} can be rewritten:

$$\dot{q}_i^{n+1} = \alpha_i^n + \ddot{q}_i^{n+1} \frac{\Delta T}{2} \quad (\text{B.10})$$

$$q_i^{n+1} = \psi_i^n + \ddot{q}_i^{n+1} \frac{(\Delta T)^2}{6} \quad (\text{B.11})$$

Substitute the above results for \dot{q}_i^{n+1} and q_i^{n+1} into the equations of motion

$$[M] \{ \ddot{q} \} + C \{ \dot{q} \} + [K] \{ q \} = \{ F \} \quad (\text{B.12})$$

$$[M] \{ \ddot{q} \}^{n+1} + [C] \{ \alpha \}^n + \frac{\Delta T}{2} [C] \{ \ddot{q} \}^{n+1} + [K] \{ \psi \}^n + \frac{(\Delta T)^2}{6} [K] \{ \ddot{q} \}^{n+1} = \{ P \}^{n+1} \quad (B.13)$$

Rearranged, $\left[[M] + \frac{\Delta T}{2} [C] + \frac{\Delta T^2}{6} [K] \right] \{ \ddot{q} \}^{n+1} = \{ P \}^{n+1} - [C] \{ \alpha \}^n - [K] \{ \psi \}^n \quad (B.14)$

Rewritten, $\overline{[M]}^{n+1} \{ \ddot{q} \}^{n+1} = \overline{\{ P \}}^{n+1} \quad (B.15)$

where $\{ P \}^{n+1} = \{ F \}$ evaluated at time $n+1$

$$\overline{[M]}^{n+1} = [M] + \frac{\Delta T}{2} [C] + \frac{\Delta T^2}{6} [K] \quad (B.16)$$

$$\overline{\{ P \}}^{n+1} = \{ P \}^{n+1} - [C] \{ \alpha \}^n - [K] \{ \psi \}^n \quad (B.17)$$

This results in the accelerations $\{ \ddot{q} \}^{n+1}$ expressed in terms of the known system values at the prior time step, $t = t^n$.

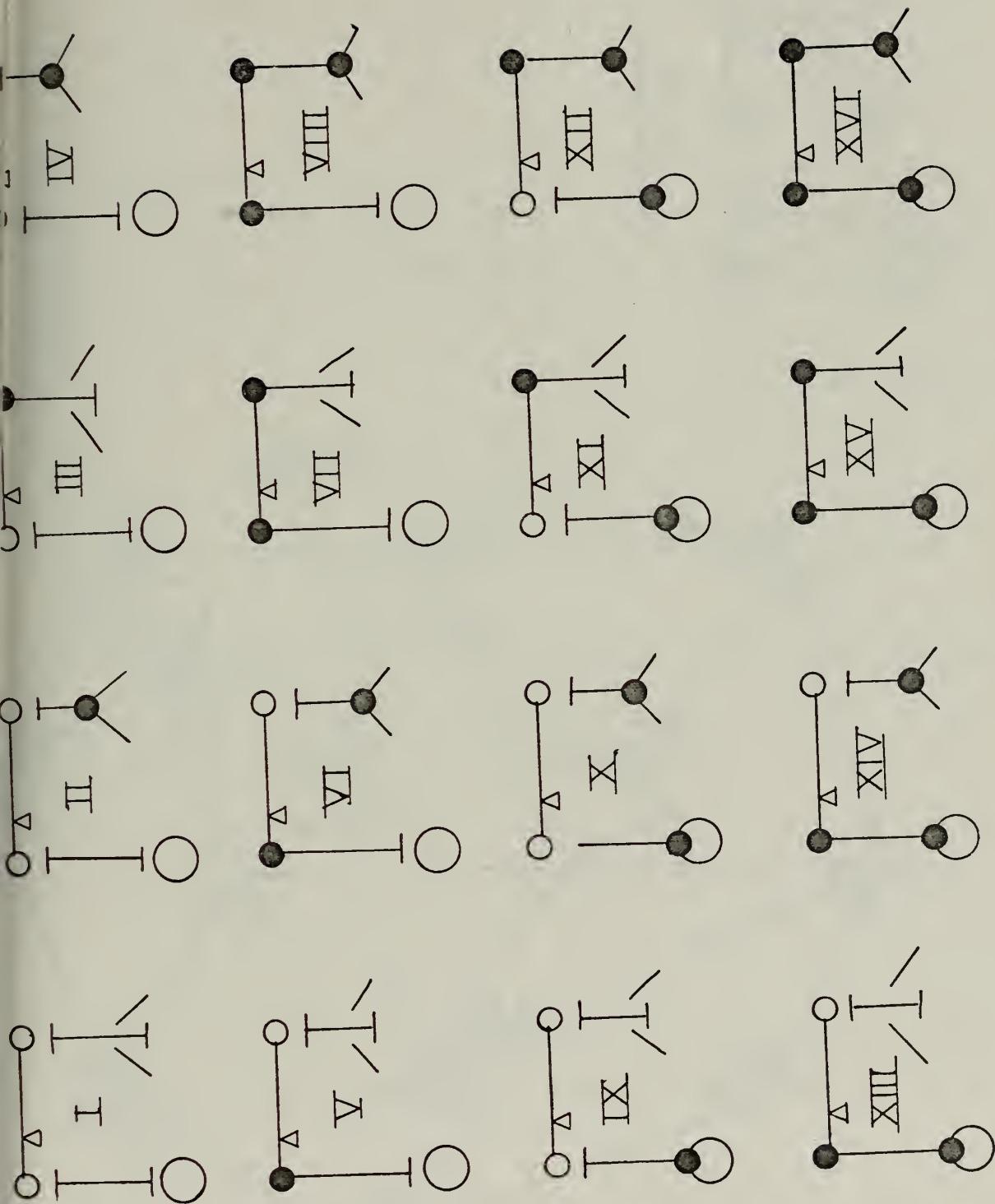
$$\{ \ddot{q} \}^{n+1} = \left[\overline{[M]}^{n+1} \right]^{-1} \overline{\{ P \}}^{n+1}$$

APPENDIX C

FOUR CLEARANCE PROBLEM

During the early stages of this study, limited research was conducted on the four clearance problem to better understand the necessary requirements of program generalization procedures. This appendix will present the unique four clearance number scheme that must be used in conjunction with the shifting algorithm presented in section 3.D. Figure C.1 indicates the configuration number associated with the various combinations of clearances, for a given time, that the model is physically permitted to take. Figure C.2 shows the resulting flow diagram. The shift lines relate the flow for a nonpositive force at the clearance indicated. Non-positive displacement differences are along the same shift lines, but in the direction opposite the force arrow.

Figure C.1 FOUR CLEARANCE SYSTEM CONFIGURATIONS



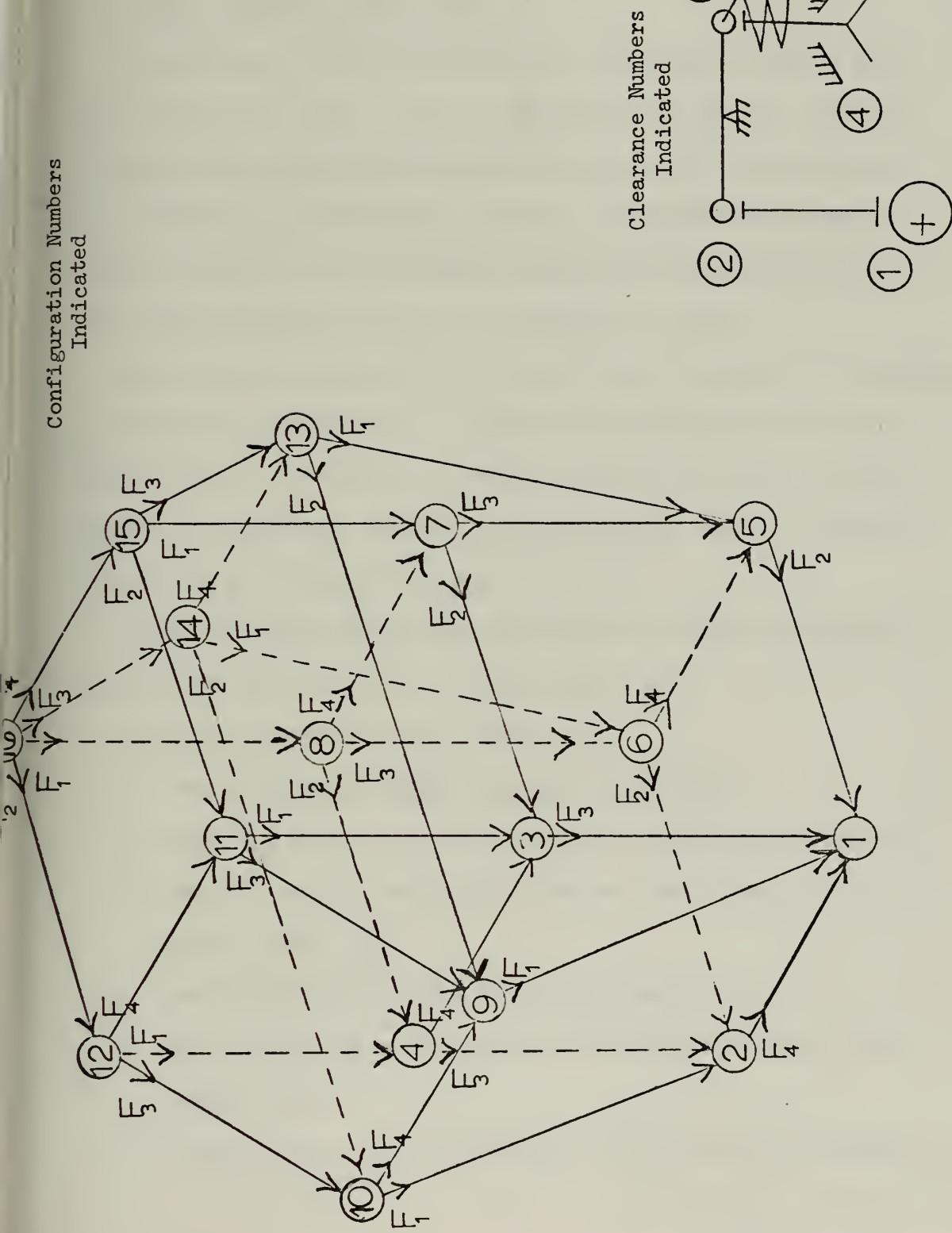


Figure C.2 4 Clearance Configuration Flow Diagram

APPENDIX D

COMPUTER CONTROL CARDS

A. IBM-360 COMPUTER CONTROL CARDS

As previously noted subroutines were complied and placed on a 2314 Disk Storage Device. This accomplished two goals. First, it eliminated the requirement to physically read into the computer a large number of punched cards. Second, it significantly reduced program execution time by greatly reducing the FORT step. This allowed the program to execute in a higher job class.

The following figures list the control cards required to accomplish various disk manipulations. A complete explanation will not be attempted here, but merely a listing of procedures that have been found to be effective. Section 3.4 of the Users Manual, reference (13), should be reviewed for more information.

In the following example control cards, the data set name is S0274. NILKA and is placed on public disk Mary.

1. Initialize disk space, Figure D.1.
2. Load subroutine NEWMRK onto disk, Figure D.2.
3. List disk data set members and space remaining, Figure D.3.
4. Execute program reading only control cards, main, and data cards, Figure D.4.
5. Load main program, CWG, onto disk, Figure D.5.
6. Execute program reading only control cards and data cards, Figure D.6.
7. Compress disk area when available space has been exceeded.

- a. Read disk members onto scratch disk without intermediate spacing, Figure D.7.
- b. Copy all members to users disk, Figure D.8.
- c. Copy all members except NEWMRK to users disk, Figure D.9.

```
// JOB CARD (GREEN CARD)
// EXEC FORTHCL,PARM.LINK='NCAL,LET'
//FORT.SYSIN DD *
--- SUBROUTINE NEWMRK CARDS ---
      STEP DELIMINATOR /* CARD
//LINK.SYSLMOD DD UNIT=2314,VOL=SER=MARY,DSN=S0274.NILKA,
//      DISP=(NEW,KEEP),SPACE=(CYL,(6,1,6)),LABEL=EXPDT=71365
//LINK.SYSIN DD *
      NAME    NEWMRK(R)
END OF JOB /*          ( ORANGE CARD )
```

Figure D.1 Initialize Disk Space

```
// JOB CARD (GREEN CARD)
// EXEC FORTHCL,PARM.LINK='NCAL,LET'
//FORT.SYSIN DD *
--- SUBROUTINE NEWMRK CARDS ---
      STEP DELIMINATOR /* CARD
//LINK.SYSLMOD DD UNIT=2314,VOL=SER=MARY,DSN=S0274.NILKA,
//      DISP=OLD,SPACE=(CYL,(6,1,6))
//LINK.SYSIN DD *
      NAME    NEWMRK(R)
END OF JOB /*          ( ORANGE CARD )
```

Figure D.2 Load Subroutine onto Disk

```
// JOB CARD (GREEN CARD)
//LIST EXEC PGM=IEHLIST,REGION=58K
//SYSPRINT DD SYSOUT=A,SPACE=(TRK,(1,1))
//DD1 DD UNIT=2314,VOL=SER=MARY,DISP=OLD
//DD2 DD UNIT=2314,VOL=SER=MARY,DISP=OLD
//SYSIN DD *,DCB=BLKSIZE=80
      LISTVTOC FORMAT,VOL=2314=MARY,DSNAME=(S0274.NILKA)
      LISTPDS DSNAME=S0274.NILKA,VOL=2314=MARY
END OF JOB /*          ( ORANGE CARD )
```

Figure D.3 List Data Set VTOC and PDS


```

// JOB CARD (GREEN CARD)
// EXEC FORTHCLG,REGION.GO=250K
//FORT.SYSIN DD *
--- MAIN PROGRAM CARDS ---
    STEP DELIMINATOR /* CARD
//LINK.PLIB DD DSN=S0274.NILKA,VOL=SER=MARY,DISP=SHR,UNIT=2314
//LINK.SYSIN DD *
INCLUDE PLIB(BRANCH,CAM,DEBUG,DEL,DRUP,DVVS,MCK)
INCLUDE PLIB(INITAL,MATRIX,NEWMRK,OLPLOT,SHIFT)
INCLUDE PLIB(BKSTEP)
INCLUDE PLIB(INPUT)
INCLUDE PLIB(GRAPH)
INCLUDE PLIB(AGT)
INCLUDE PLIB(THESIS)
INCLUDE PLIB(PRINT)
INCLUDE PLIB(ARRAY)
    STEP DELIMINATOR /* CARD
//GO.FT06F001 DD SYSOUT=A,SPACE=(CYL,(4,2))
//GO.SYSIN DD *
--- DATA CARDS ---
END OF JOB /*          ( ORANGE CARD )

```

Figure D.4 Execute Program with Subroutines on Disk

```

// JOB CARD (GREEN CARD)
// EXEC FORTHCL,PARM.FORT='NAME=CWG,LOAD,MAP',
//      PARM.LINK='LET,NCAL,MAP,XREF'
//FORT.SYSIN DD *
--- MAIN PRGRAM CARDS ---
    STEP DELIMINATOR /* CARD
//LINK.SYSLMOD DD UNIT=2314,VOL=SER=MARY,DSN=S0274.NILKA,
//      DISP=OLD,SPACE=(CYL,(6,1,6))
//LINK.SYSIN DD *
NAME CWG(R)
END OF JOB /*          ( ORANGE CARD )

```

Figure D.5 Load Main Program onto Disk

```

// JOB CARD (GREEN CARD)
// EXEC LGOP,REGION.GO=250K
//LINK.PLIB DD UNIT=2314,VOL=SER=MARY,DSN=S0274,NILKA,DISP=SHR
//LINK.SYSIN DD *
INCLUDE PLIB(ARRAY,BKSTEP,BRANCH,CAM,DEBUG,DEL,DRUP,DVVS,GRAPH,MCK)
INCLUDE PLIB(INITAL,INPUT,MATRIX,NEWMRK,OLPLOT,PRINT,SHIFT,THESIS)
INCLUDE PLIB(CWG)
ENTRY CWG
    STEP DELIMINATOR /* CARD
//GO.SYSIN DD *
--- DATA CARDS ---
END OF JOB /*          ( ORANGE CARD )

```

Figure D.6 Execute Entire Program from Disk


```

// JOB CARD (GREEN CARD)
//STEP1 EXEC PGM=IEHLIST
//SYSPRINT DD SYSOUT=A
//D5 DD UNIT=2314, VOL=(PRIVATE,,SER=MARY), DISP=SHR
//SYSIN DD *
    LISTVTOC VOL=2314=MARY
//STEP2 EXEC PGM=IEHPROGM
//SYSPRINT DD SYSOUT=A
//SPOOL3 DD DISP=OLD, UNIT=2314, VOLUME=SER=SPOOL3
//TMPLIB DD DISP=(NEW,KEEP), UNIT=2314, DSNAME=NILKA.TMPLIB,
//          SPACE=(CYL,(6,0,17)), DCB=(RECFM=U, BLKSIZE=7294),
//          VOLUME=SER=SPOOL3
//SYSIN DD *
    SCRATCH DSNAME=S0274.NILKA, VOL=2314=SPOOL3, PURGE
    RENAME DSNAME=NILKA.TMPLIB, VOL=2314=SPOOL3, NEWNAME=S0274.NILKA
    STEP DELIMINATOR /* CARD
//STEP3 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DISP=OLD, UNIT=2314, DSNAME=S0274.NILKA, VOLUME=SER=MARY
//SYSUT2 DD DISP=OLD, UNIT=2314, DSNAME=S0274.NILKA, VOLUME=SER=SPOOL3
//SYSIN DD *
    COPY
END OF JOB /*           ( ORANGE CARD )

```

Figure D.7 Read Data Set Members onto Scratch Disk

```

// JOB CARD (GREEN CARD)
//STEP3 EXEC PGM=IEHPROGM
//SYSPRINT DD SYSOUT=A
//MARY DD DISP=OLD, UNIT=2314, VOLUME=SER=MARY
//SYSIN DD *
    SCRATCH DSNAME=S0274.NILKA, VOL=2314=MARY, PURGE
    STEP DELIMINATOR /* CARD
//STEP4 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DISP=CLD, UNIT=2314, DSNAME=S0274.NILKA, VOLUME=SER=SPOOL3
//SYSUT2 DD DISP=(NEW,KEEP), UNIT=2314, DSNAME=S0274.NILKA,
//          SPACE=(CYL,(6,1,6)), DCB=(RECFM=U, BLKSIZE=7294),
//          VOLUME=SER=MARY, LABEL=EXPDT=71365
//SYSIN DD *
    COPY
END OF JOB /*           ( ORANGE CARD )

```

Figure D.8 Copy all Data Set Members

```

// JOB CARD (GREEN CARD)
//STEP3 EXEC PGM=IEHPROGM
//SYSPRINT DD SYSOUT=A
//MARY DD DISP=OLD, UNIT=2314, VOLUME=SER=MARY
//SYSIN DD *
    SCRATCH DSNAME=S0274.NILKA, VOL=2314=MARY, PURGE
    STEP DELIMINATOR /* CARD
//STEP4 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DISP=OLD, UNIT=2314, DSNAME=S0274.NILKA, VOLUME=SER=SPOOL3
//SYSUT2 DD DISP=(NEW,KEEP), UNIT=2314, DSNAME=S0274.NILKA,
//          SPACE=(CYL,(6,1,6)), DCB=(RECFM=U, BLKSIZE=7294),
//          VOLUME=SER=MARY, LABEL=EXPDT=71365
//SYSIN DD *
    COPY TYPCOPY=E, MAXNAME=1
    MEMBER NAME=(NEWMRK)
END OF JOB /*           ( ORANGE CARD )

```

Figure D.9 Copy Part of Data Set Members

B. SDS-9300 COMPUTER CONTROL CARDS

Transferring the SDS-9300 program, in Fortran source code, from punched cards to magnetic tape is a relatively easy task. Once accomplished, repeated compiling and execution of the program is made much easier. Figure D.10 indicates the control cards required to read the punched cards onto magnetic tape.

```
SDS-9300 COMPUTER BINARY BOOT CARD*
-JOB
-LOAD X, TAPEDT
$+
--- MAIN PROGRAM CARDS ---
=EOF
---- SUBROUTINE BAR1 CARDS ---
---- SUBROUTINE BAR2 CARDS ---
---- SUBROUTINE BAR3 CARDS ---
---- SUBROUTINE BAR4 CARDS ---
=EOF
---- SUBROUTINE CAM CARDS ---
=EOF
ETC. UNTIL ALL SUBROUTINES ARE READ.
=EOF
```

*Indicates card supplied by Computer Laboratory

Figure D.10 Transferring FORTRAN Source Code From Cards to Magnetic Tape

The computer laboratory provided routine TAPEDT enables the user to change the program coding with a minimum of cards, while the program remains on the tape.

Figure D.11 shows the required control cards to compile, load, and execute the magnetic tape Fortran source program.


```
SDS-9300 COMPUTER BINARY BOOT CARD*
JOB
$ASSIGN SI=MT2A
$REWIND SI
$DATE 17,DEC,1971
$TITLE .           CHARLES W. GNILKA -- THESIS --
$AGT
$ASSIGN 4=MT4A
$LABEL MAIN
$FORTRAN GO
$LABEL CAM
$FORTRAN GO
$LABEL MIDBAR
$FORTRAN GO
$LABEL SPRCTR
$FORTRAN GO
$LABEL BAR
$FORTRAN GO
$LABEL DISPLAY
$FORTRAN GO
$LABEL INITIAL
$FORTRAN GO
$LABEL READ
$FORTRAN GO
$LOAD XR,MAP
$SEG MAIN-BAR-CAM-(MIDBAR,SPRCTR,DISPLAY,INITIAL,READ)
$DATA
```

*Indicates Card Supplied by Computer Laboratory

Figure D.11 Program Execution From FORTRAN Object on Magnetic Tape

To place the program on magnetic tape as binary executable code, the control cards are as in Figure D.11 with an additional control card immediately after the Δ JOB card. Figure D.12 indicates the required control cards.


```
SDS-9300 COMPUTER BINARY BOOT CARD*
-JOB
$-
→ASSIGN GO=MT3A
( THE REMAINING CONTROL CARDS ARE EXACTLY AS GIVEN ABOVE )
```

Figure D.12 Placing Program as Binary Code on Magnetic Tape

Once the binary code is on magnetic it may be executed approximately 75% faster than in uncompiled FORTRAN source form. The control cards to accomplish this are shown in Figure D.13

```
SDS-9300 COMPUTER BINARY BOOT CARD*
-JOB
$-
→AGT
→ASSIGN BI=MT3A
→ASSIGN GO=BI
→ASSIGN SI=MT3A
→LLOAD XR,MAP
→SEG MAIN-BAR-CAM-(MIDBAR,SPRCTR,DISPLY,INITAL,READ)
→DATA
```

Figure D.13 Program Execution From Binary Code on Magnetic Tape

APPENDIX E

IBM-360 COMPUTER PROGRAM DATA CARDS

GNILKA, C. W. 17 DECEMBER THREE CLEARANCES NO. 1

1,000 RPM VALVE SEAT

1000. RPM

.7 NO CLEARANCE AT VALVE SEAT AND PUSHROD/ROCKER

ES PRINT OUT MATRICES MPRINT

0 CALCOMP GRAPHS DESIRED

0 AGT OUTPUT

0 NO PUNCHED CARDS; MAGNETIC TAPE

.052 DEGREES TO PRINT DEBUG OUTPUT OR STOP

0 10 PRINTED OUTPUT DESIRED; NCOUNT

10 PRINTED OUTPUT DESIRED FOR DEBUG PURPOSES - IPRINT

10 OFFLINE PRINTER GRAPH

1 IDEG -- FORCED TO STAY IN CONFIG 7

0.000001 DELT- - TIME INCREMENT

DISPL. DIFF. BETWEEN ROD & ROCKER ** Q(2)-Q(8)

DISPL. DIFF. BETWEEN CAM & ROD ** Q(1)-Q(9)

CAM PROFILE ** Q(9)

DISPL. OF LOWER END OF PUSHROD ** Q(1)

FORCE ON PUSHROD ** F(1)

AXIAL OSCILLATIONS OF THE PUSHROD ** Q(8)-Q(1)

DISPLACEMENT OF VALVE ** Q(7)

POSITION OF VALVE SEAT ** Q(10)

FORCE ON ROCKER ** F(2)

AXIAL OSCILLATIONS OF THE VALVE STEM

FORCE ON VALVE ** F(7)

CAM PROFILE AND DISPL. OF PUSHROD

DISPL. OF VALVE AND VALVE SEAT

DEGREES OF CAM ROTATION

APPENDIX F

This appendix presents two additional examples for comparison with the 11,000 RPM examples presented in chapter 6.C.

A. EXAMPLE 3, THREE CLEARANCES, 9,000 RPM, VALVE SEAT

Approximately 800 degrees of cam rotation is indicated on the associated graphs. These graphs only indicate the indicate the major system points of interest and are not otherwise annotated. The graphs, in order of presentation are:

1. Figure F.1

a. Cam Profile, $q_9 \times 10^{-1}$

b. Displacement of lower end of pushrod

$$q_1 \times 10^{-1}$$

c. Displacement difference between cam and pushrod

$$q_1 - q_9 \times 10^{-3}$$

2. Figure F.2, Force on Pushrod

$$F_1 \times 10^2$$

3. Figure F.3, Force on Rocker

$$F_2 \times 10^2$$

4. Figure F.4, Force on Valve

$$F_7 \times 10^2$$

5. Figure F.5, Displacement of Valve

$$q_7 \times 10^{-1}$$

6. Figure F.6, Axial Oscillations of the Pushrod

$$(q_8 - q_1) \times 10^{-3}$$

7. Figure F.7, Axial Oscillation of the Valve Stem

$$(q_7 - q_4) \times 10^{-3}$$

8. Figure F.8

- a. Displacement Difference between Cam and Pushrod

$$(q_1 - q_9) \times 10^{-3}$$

- b. Displacement Difference between Pushrod and Rocker

$$(q_2 - q_8) \times 10^{-2}$$

B. EXAMPLE 3, THREE CLEARANCES 10,000 RPM, VALVE SEAT

This last example referred to in chapter 6.C. also presents approximately 800 degrees of cam rotation. These graphs only indicate the major system points of interest and are not otherwise annotated. The graphs, in order of presentation are:

1. Figure F.9

- a. Cam Profile, $q_9 \times 10^{-1}$

- b. Displacement of lower end of pushrod

$$q_1 \times 10^{-1}$$

- c. Displacement difference between cam and pushrod

$$q_1 - q_9 \times 10^{-3}$$

2. Figure F.10, Force on Pushrod

$$F_1 \times 10^2$$

3. Figure F.11, Force on Rocker

$$F_2 \times 10^2$$

4. Figure F.12, Force on Valve

$$F_7 \times 10^2$$

5. Figure F.13, Displacement of Valve

$$q_7 \times 10^{-1}$$

6. Figure F.14, Axial Oscillations of the Pushrod

$$(q_8 - q_1) \times 10^{-3}$$

7. Figure F.15, Axial Oscillation of the Valve Stem

$$(q_7 - q_4) \times 10^{-3}$$

8. Figure F.16

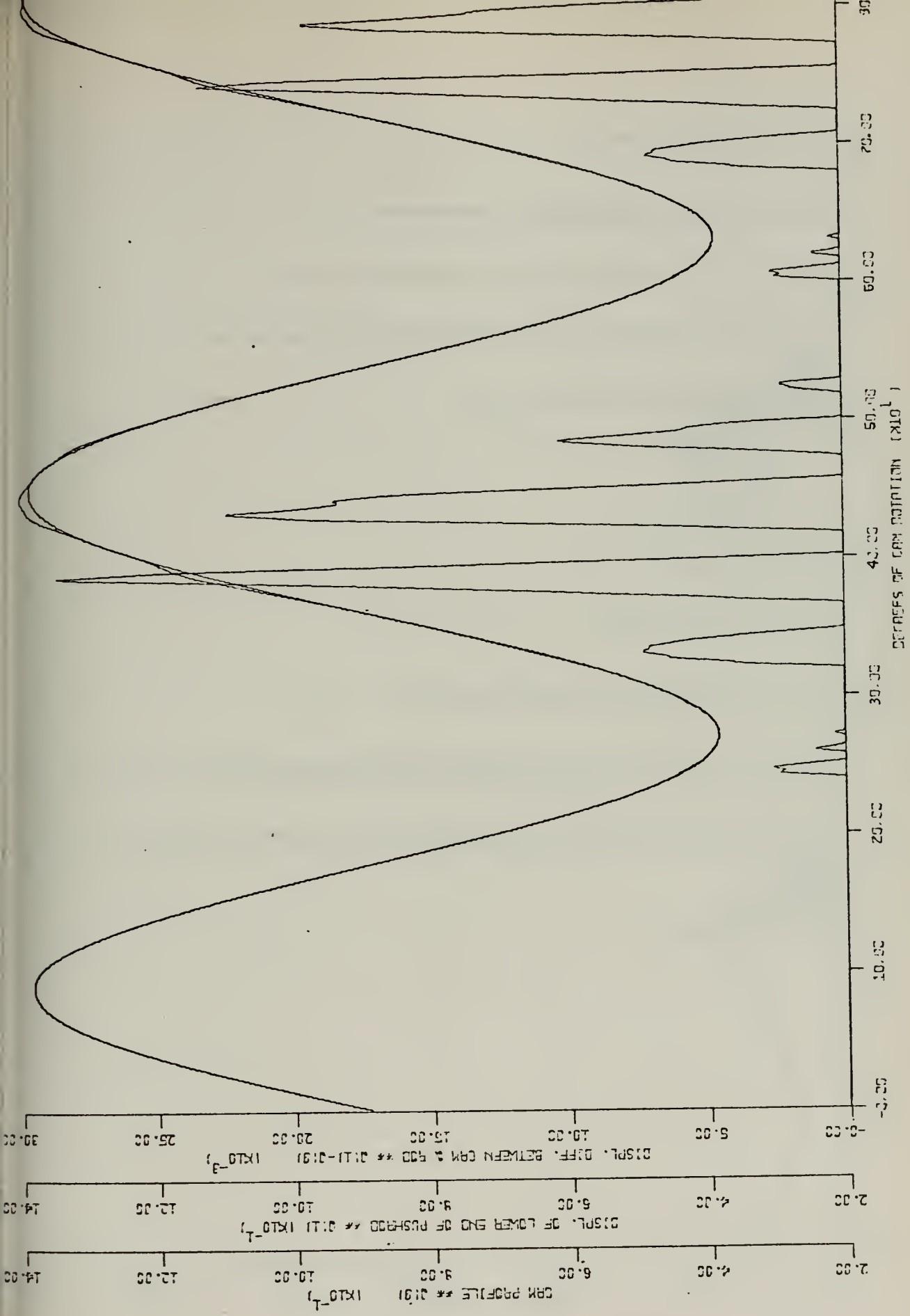
a. Displacement Difference between cam and pushrod

$$(q_1 - q_9) \times 10^{-3}$$

b. Displacement Difference between Pushrod and Rocker

$$(q_2 - q_8) \times 10^{-2}$$

Figure F.1 THREE CLEARANCES 9,000 RPM UNLUE SEAT



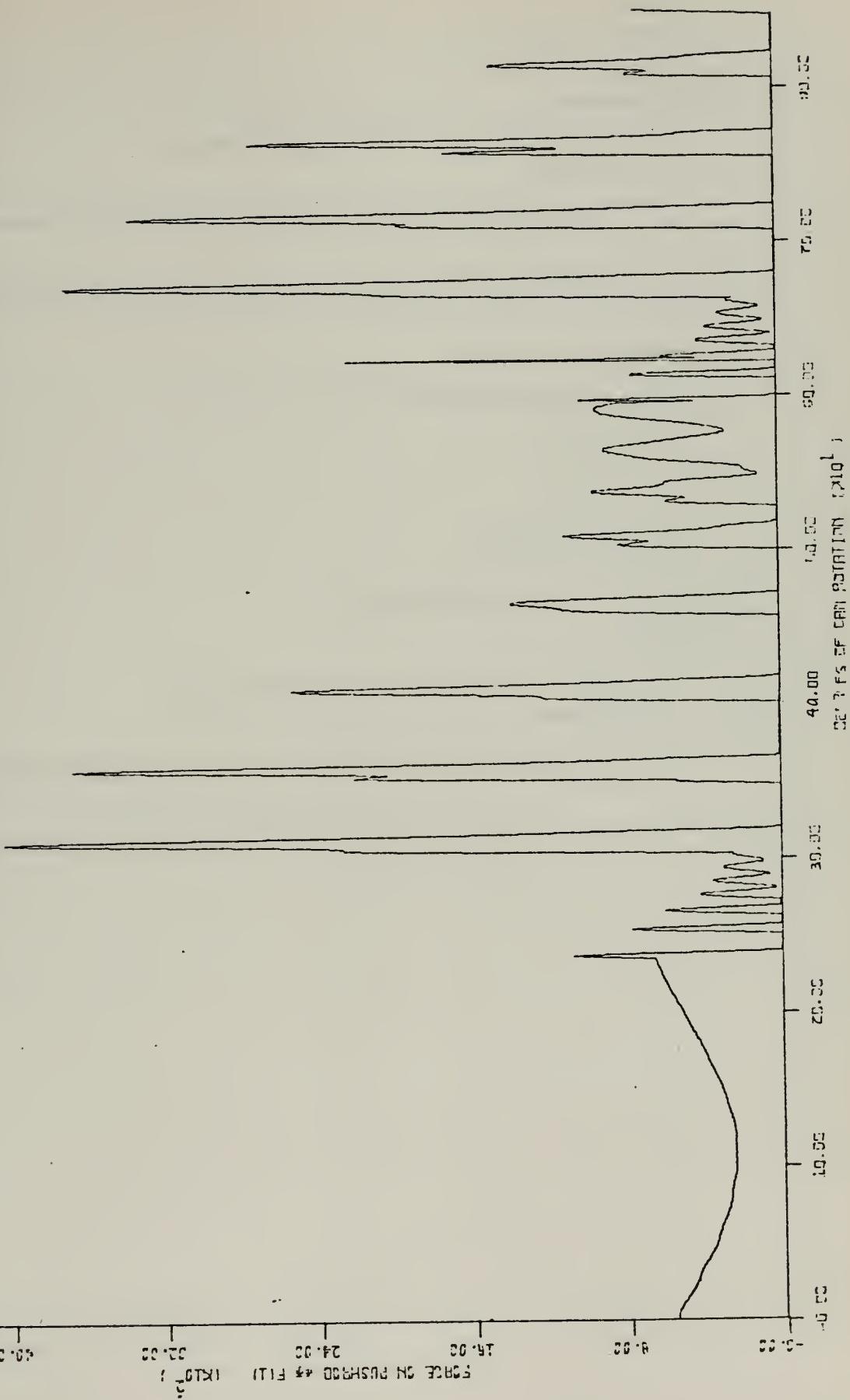


Figure F.2 THREE CLEARANCES

9,000 RPM

VALUE SETAT

Figure F.3 THREE CLEARANCES

9,000 RPM VALUE SFAT

REF. REFS OF CAN ROTATION (deg)

40.00 39.00 38.00 37.00 36.00 35.00 34.00 33.00 32.00 31.00 30.00

30.00 29.00 28.00 27.00 26.00 25.00 24.00 23.00 22.00 21.00 20.00

20.00 19.00 18.00 17.00 16.00 15.00 14.00 13.00 12.00 11.00 10.00

10.00 9.00 8.00 7.00 6.00 5.00 4.00 3.00 2.00 1.00 0.00

0.00 -1.00 -2.00 -3.00 -4.00 -5.00 -6.00 -7.00 -8.00 -9.00 -10.00

-10.00 -11.00 -12.00 -13.00 -14.00 -15.00 -16.00 -17.00 -18.00 -19.00 -20.00

-20.00 -21.00 -22.00 -23.00 -24.00 -25.00 -26.00 -27.00 -28.00 -29.00 -30.00

-30.00 -31.00 -32.00 -33.00 -34.00 -35.00 -36.00 -37.00 -38.00 -39.00 -40.00

-40.00 -41.00 -42.00 -43.00 -44.00 -45.00 -46.00 -47.00 -48.00 -49.00 -50.00

-50.00 -51.00 -52.00 -53.00 -54.00 -55.00 -56.00 -57.00 -58.00 -59.00 -60.00

-60.00 -61.00 -62.00 -63.00 -64.00 -65.00 -66.00 -67.00 -68.00 -69.00 -70.00

-70.00 -71.00 -72.00 -73.00 -74.00 -75.00 -76.00 -77.00 -78.00 -79.00 -80.00

-80.00 -81.00 -82.00 -83.00 -84.00 -85.00 -86.00 -87.00 -88.00 -89.00 -90.00

-90.00 -91.00 -92.00 -93.00 -94.00 -95.00 -96.00 -97.00 -98.00 -99.00 -100.00

-100.00 -101.00 -102.00 -103.00 -104.00 -105.00 -106.00 -107.00 -108.00 -109.00 -110.00

-110.00 -111.00 -112.00 -113.00 -114.00 -115.00 -116.00 -117.00 -118.00 -119.00 -120.00

-120.00 -121.00 -122.00 -123.00 -124.00 -125.00 -126.00 -127.00 -128.00 -129.00 -130.00

-130.00 -131.00 -132.00 -133.00 -134.00 -135.00 -136.00 -137.00 -138.00 -139.00 -140.00

-140.00 -141.00 -142.00 -143.00 -144.00 -145.00 -146.00 -147.00 -148.00 -149.00 -150.00

-150.00 -151.00 -152.00 -153.00 -154.00 -155.00 -156.00 -157.00 -158.00 -159.00 -160.00

-160.00 -161.00 -162.00 -163.00 -164.00 -165.00 -166.00 -167.00 -168.00 -169.00 -170.00

-170.00 -171.00 -172.00 -173.00 -174.00 -175.00 -176.00 -177.00 -178.00 -179.00 -180.00

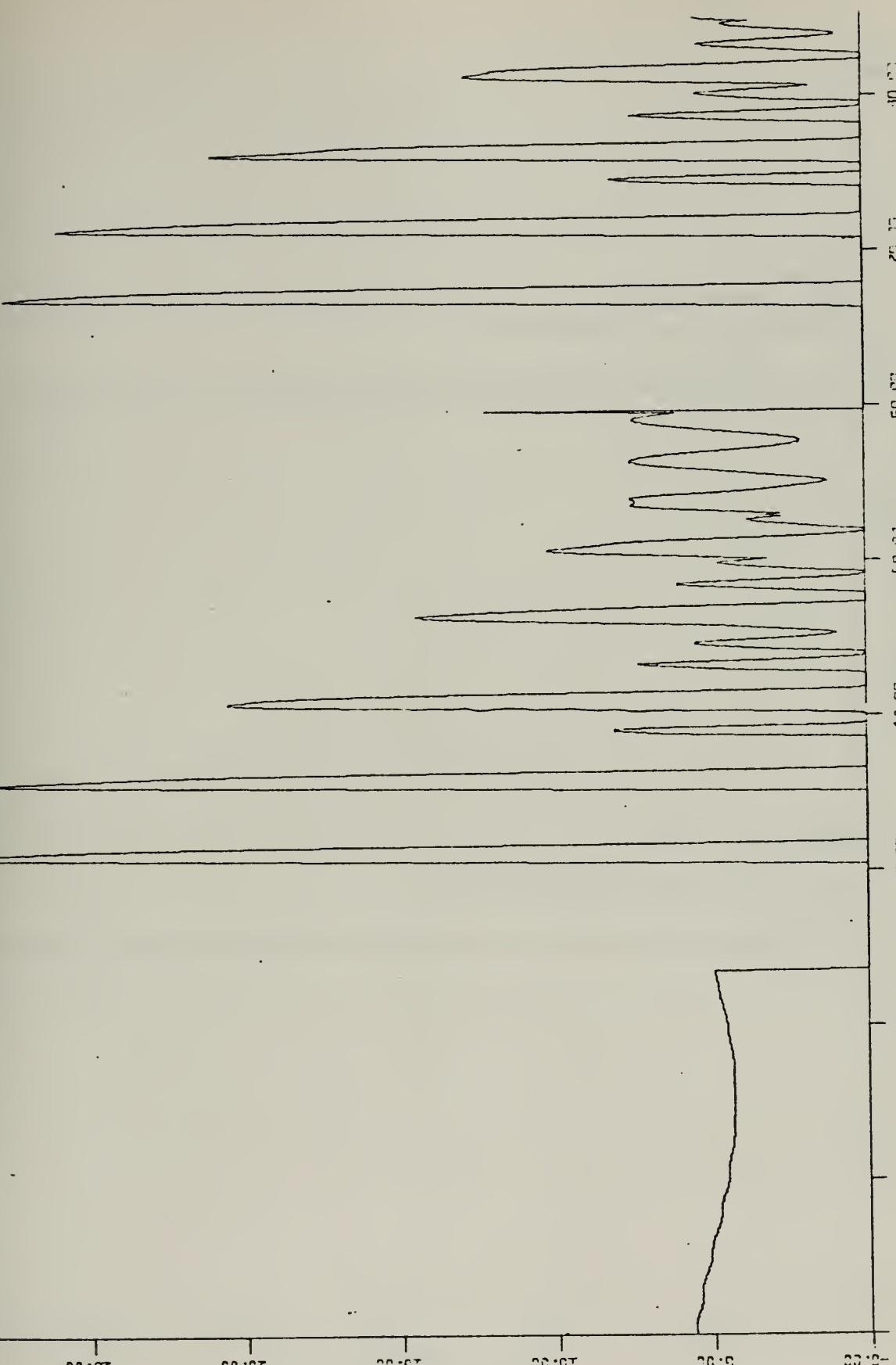


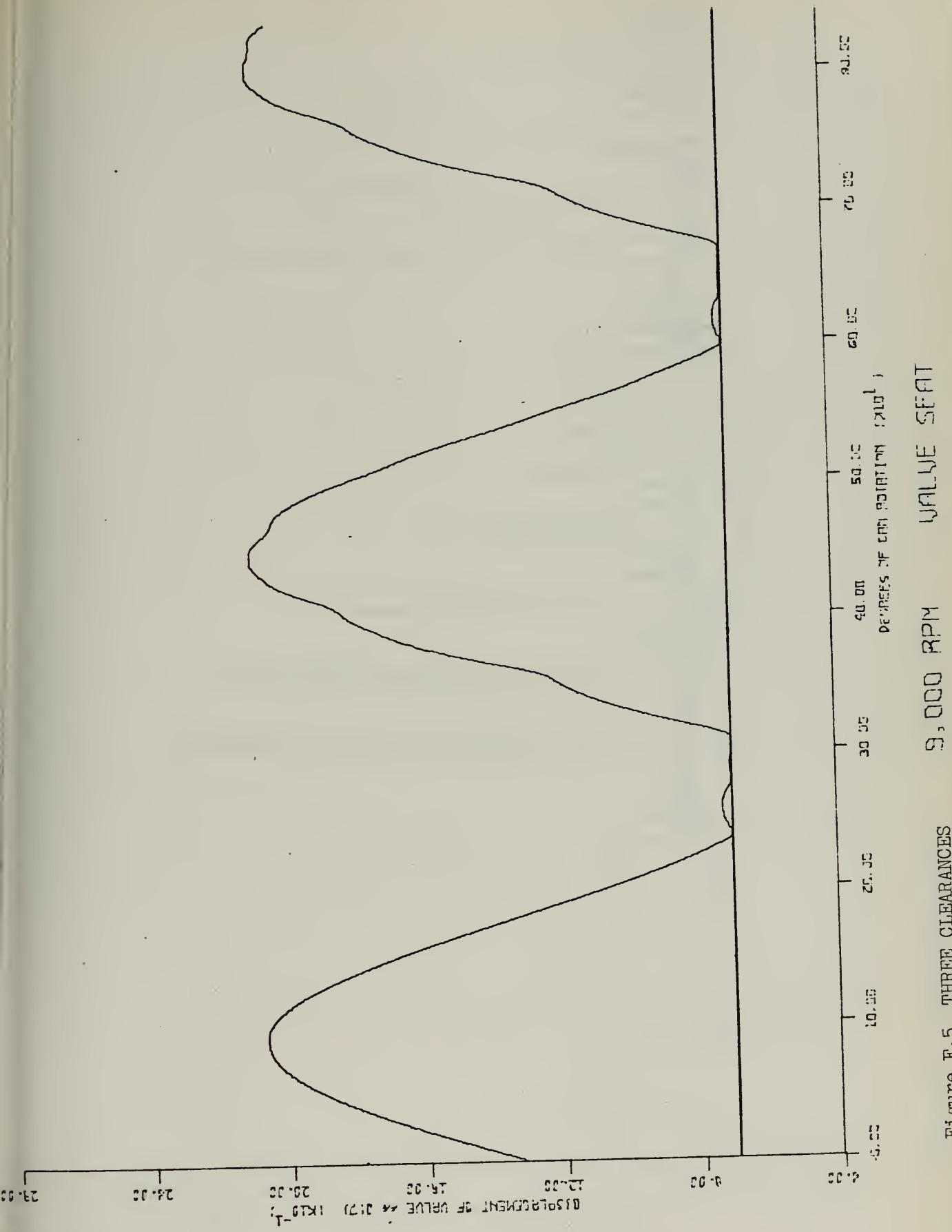
Figure F.4 THREE CLEARANCES

9,000 RPM VALUE SEAT

-3.00 10.00 20.00 30.00 40.00 50.00 60.00 70.00 80.00

DEGREES OF CAN ROTATION (X10⁻²)

FORCE ON VALUE ** E17) (KLD⁻²)



9,000 RPM VALUE SEAT

9,000 RPM

THREE CLEARANCES

Figure F.6

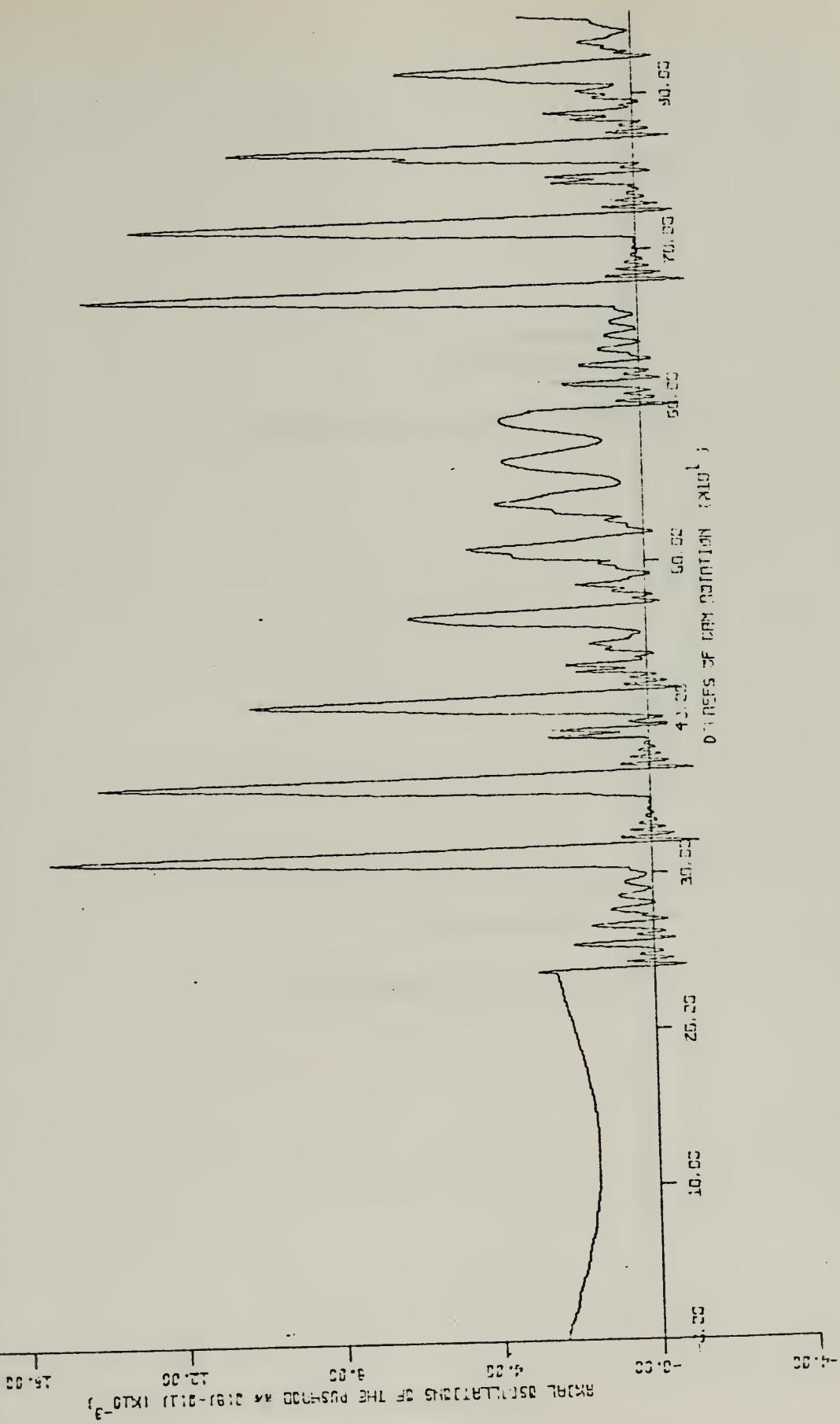


Figure F.7 THREE CLEARANCES

9,000 RPM VALUE SEAT

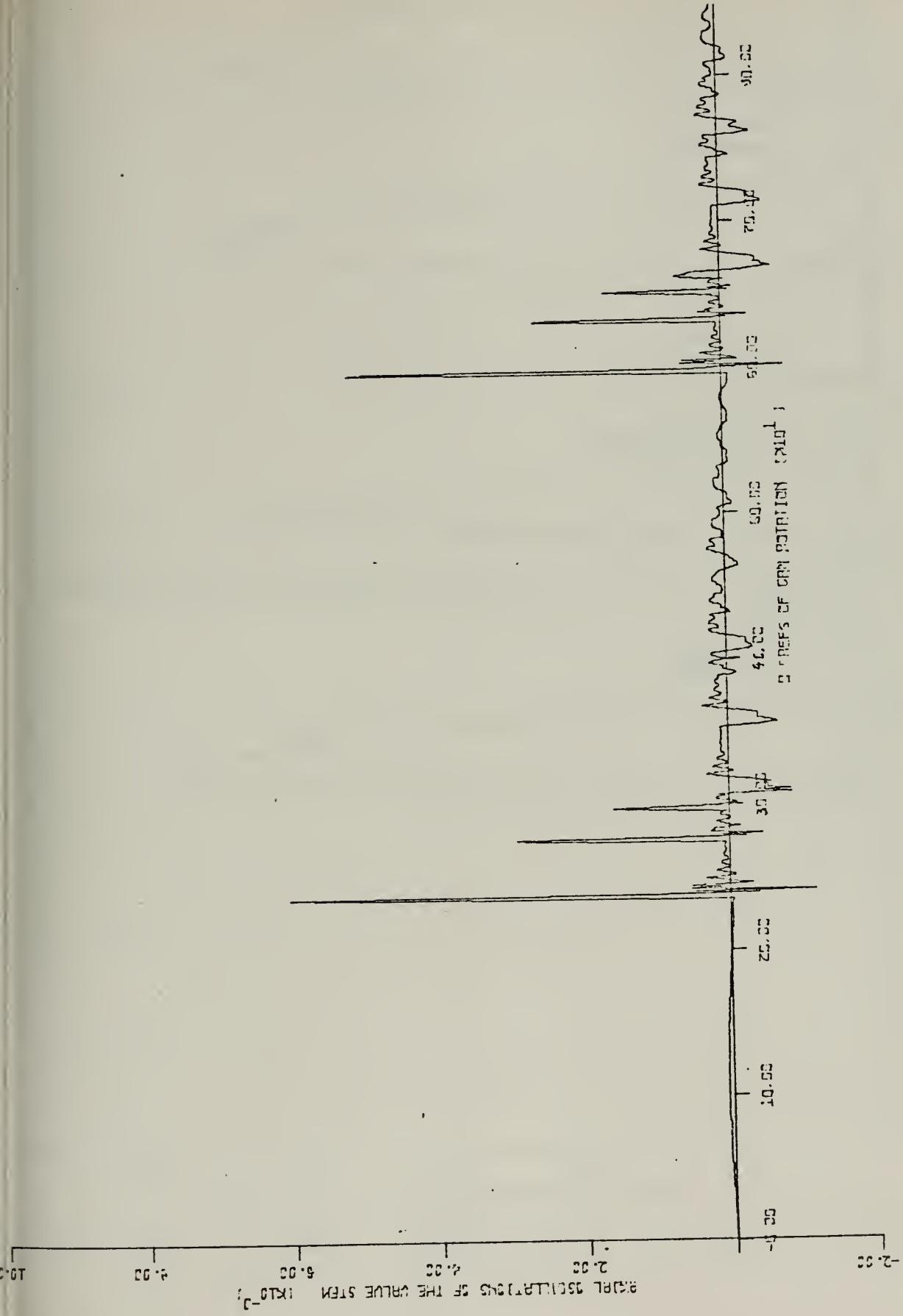


Figure F.8 THREE CLEARANCES

3,000 RPM UNLINED SEAT

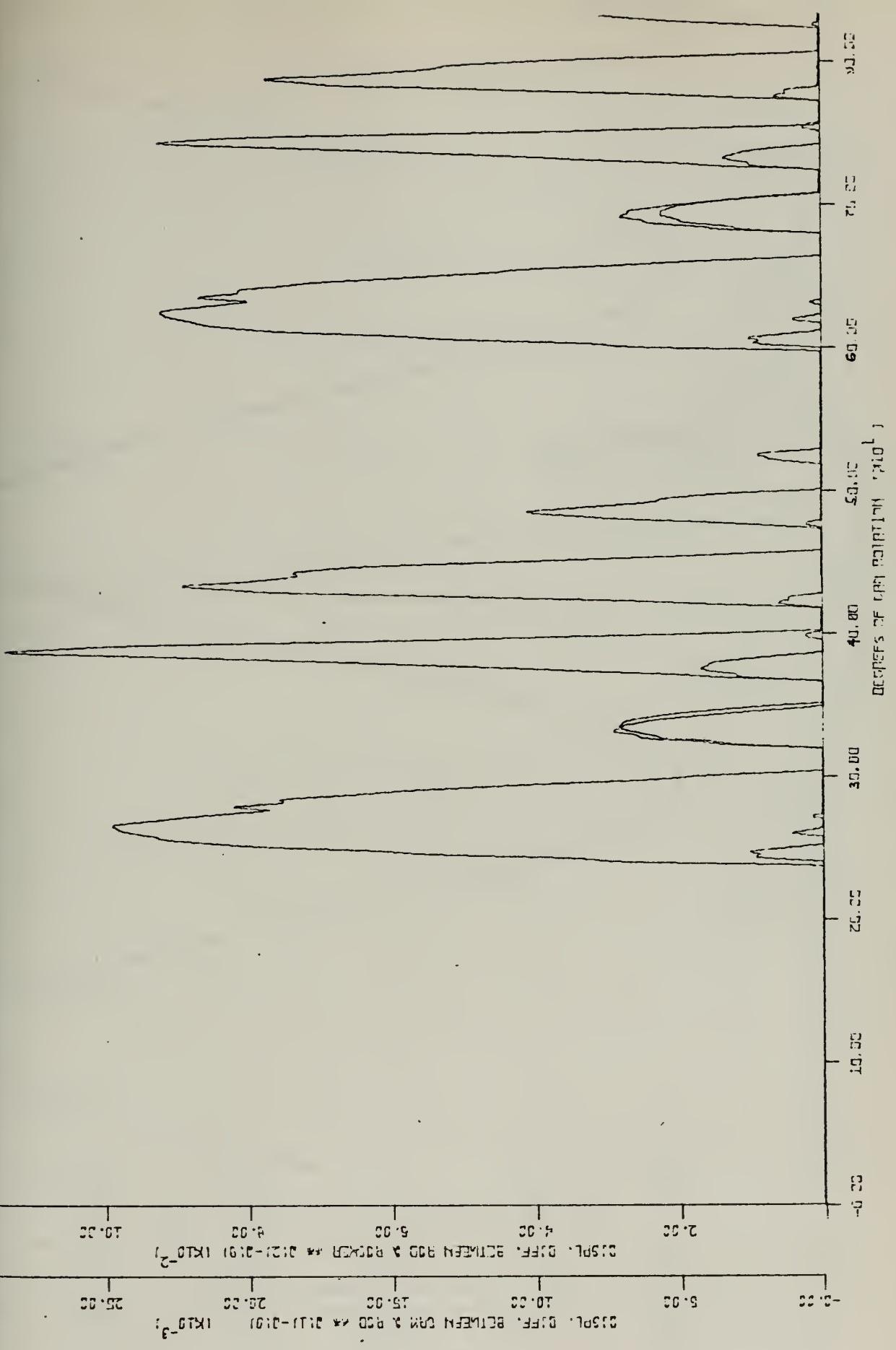


Figure F.9 THREE. CLEARANCES 10,000 RPM VALUE SEAT

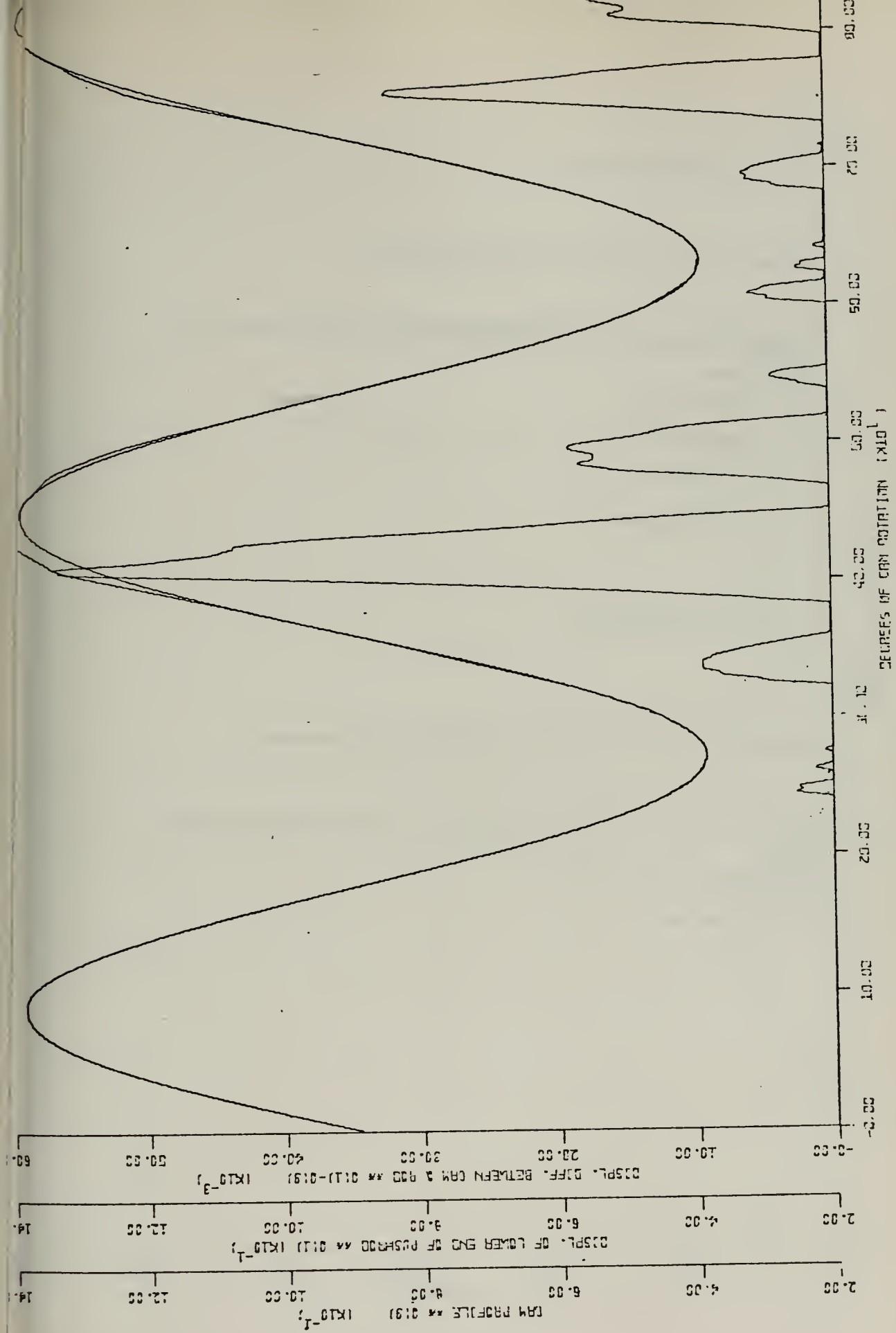


Figure F. 10 THREE CLEARANCES 10,000 RPM VALUE SEAT

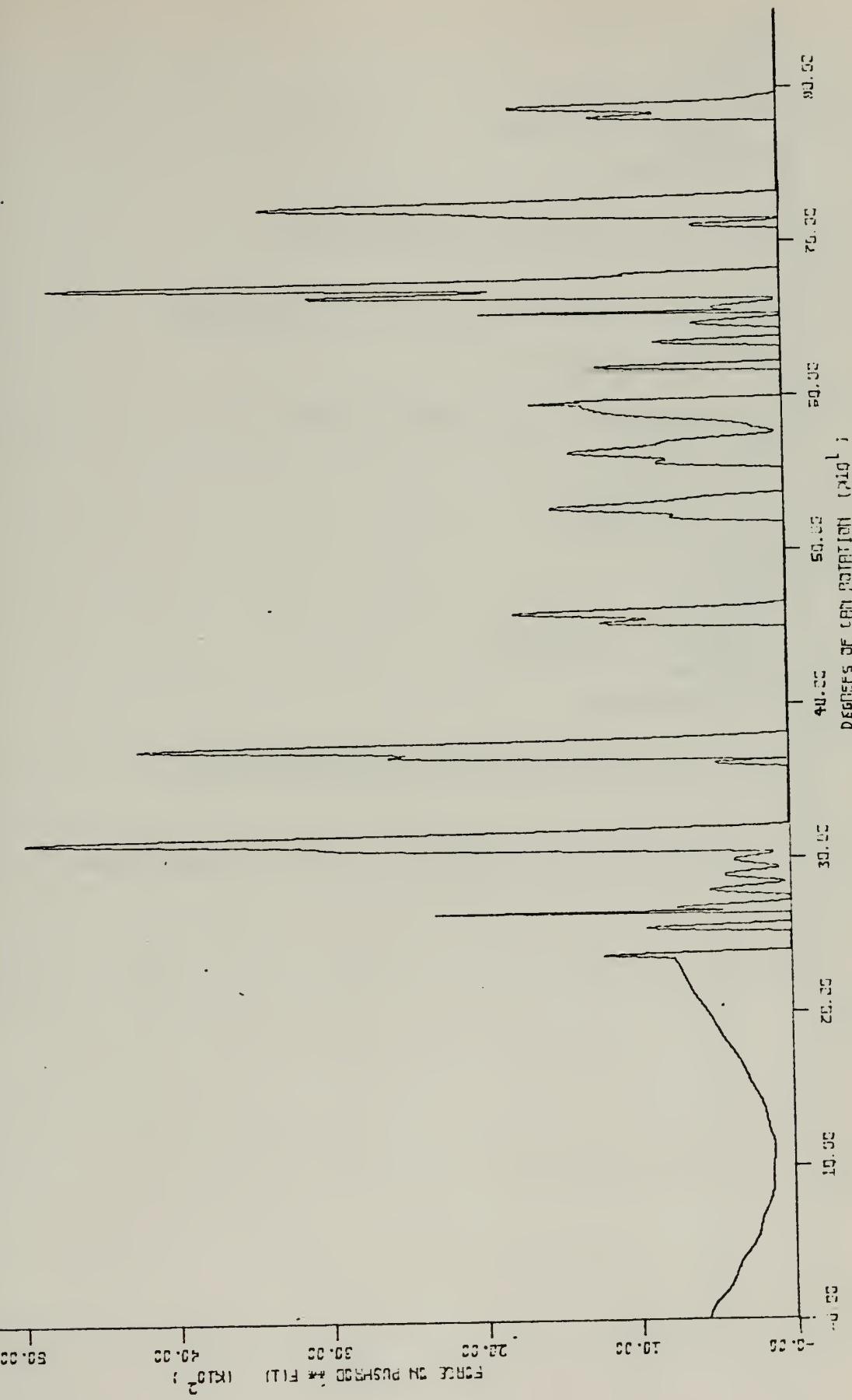


Figure F.11 THREE CLEARANCES 10,000 RPM VALUE USE SPFT

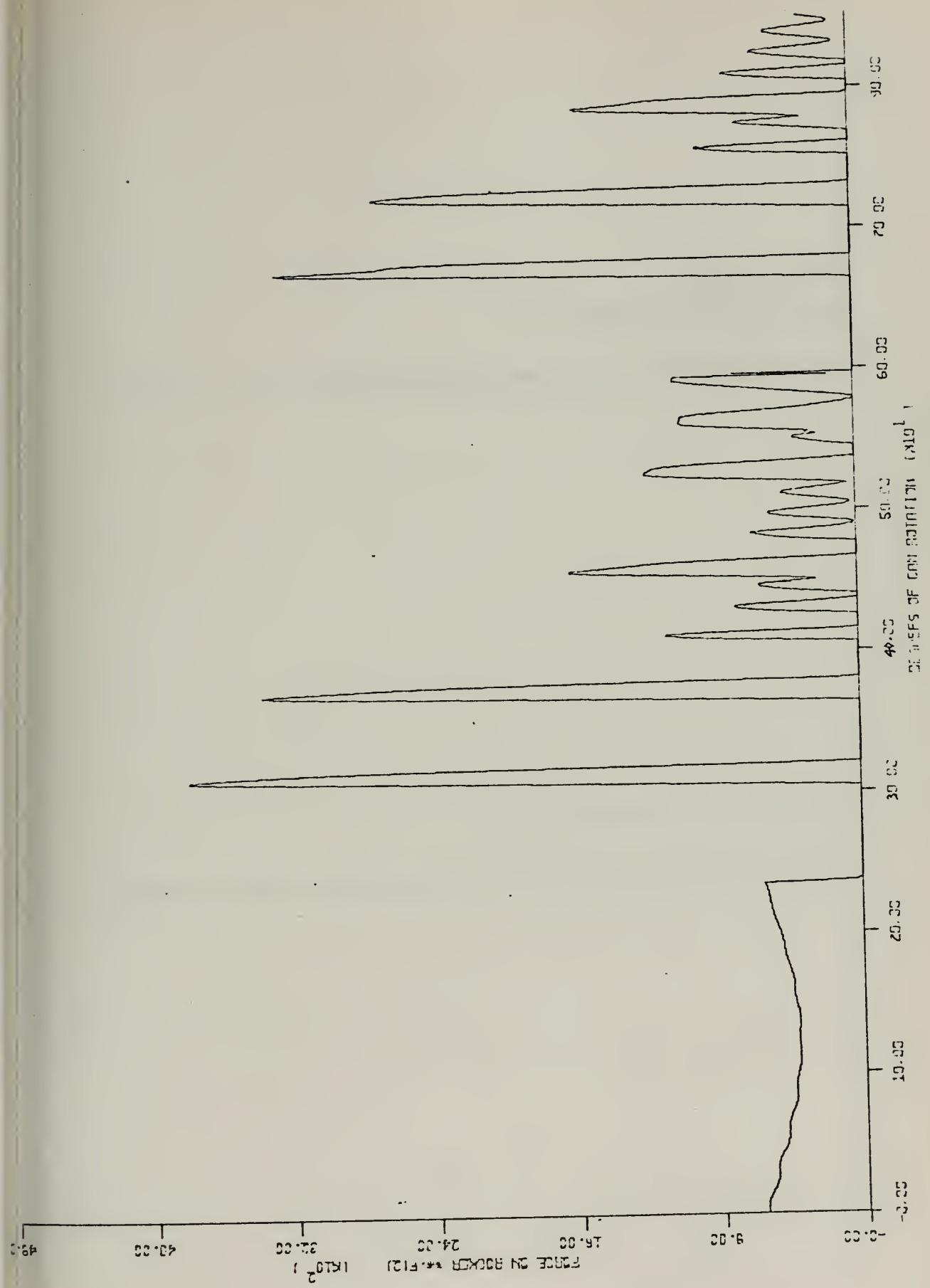


Figure F.12 THREE CLEARANCES 10,000 RPM VALUE SEAT

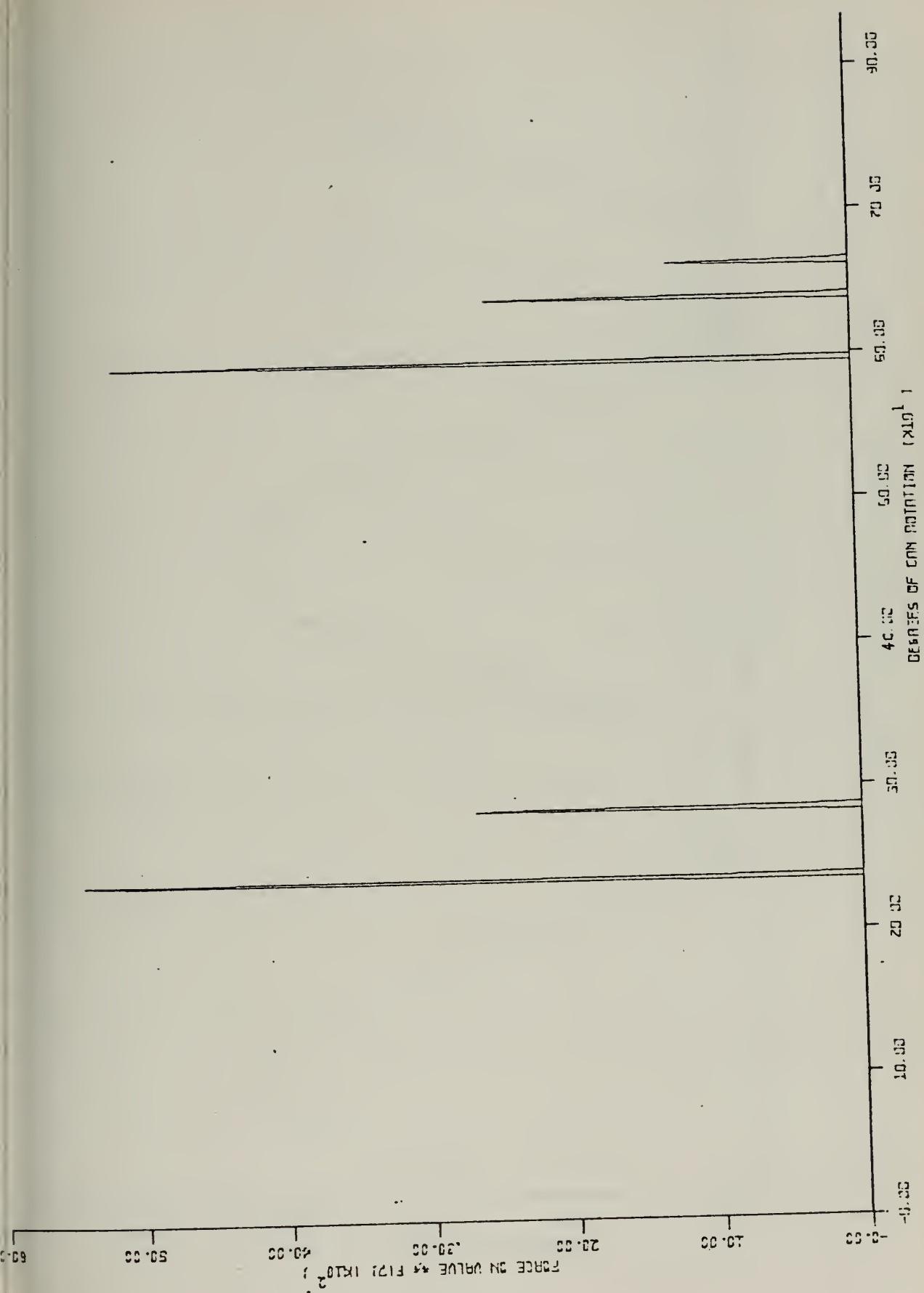


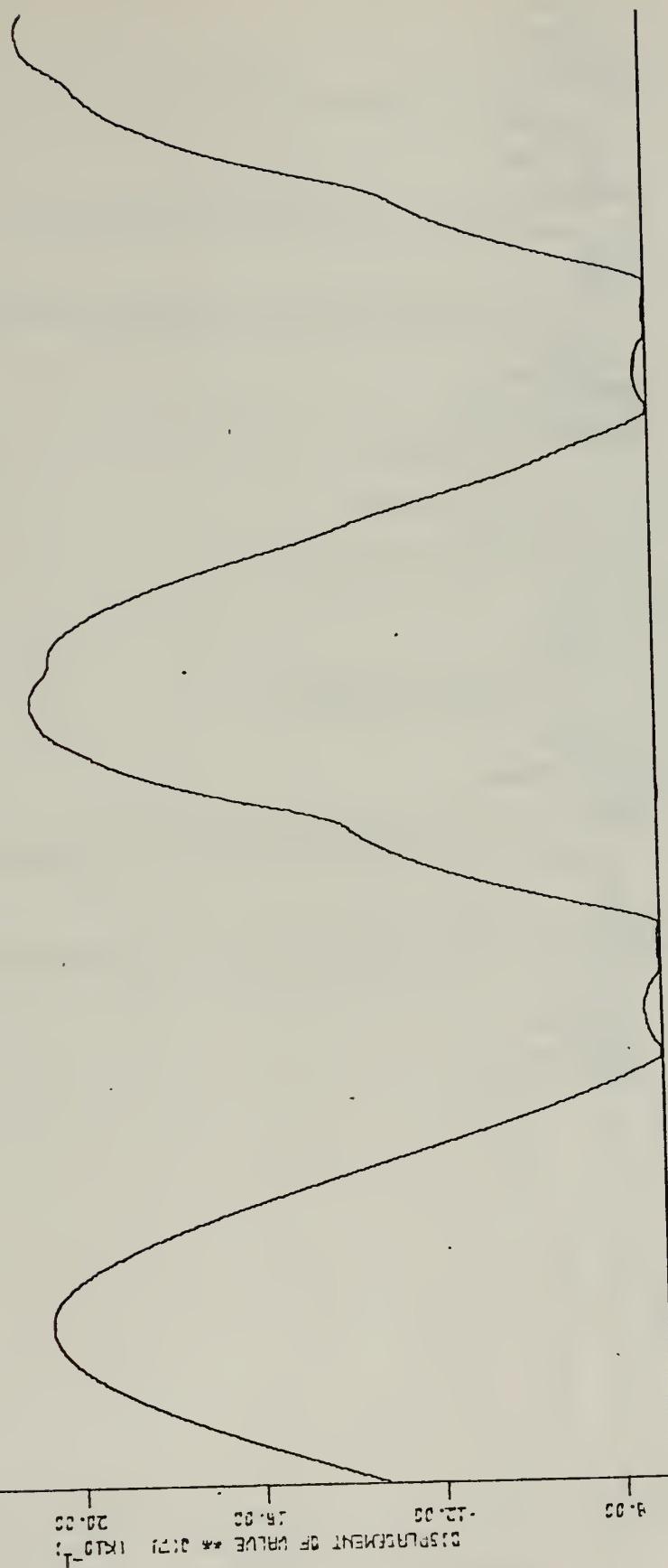
Figure F.13 THREE CLEARANCES

10,000 RPM VALUE SEAT

DEGREES OF CRANK ROTATION (deg.) ;
10.00 15.00 20.00 25.00 30.00 35.00 40.00

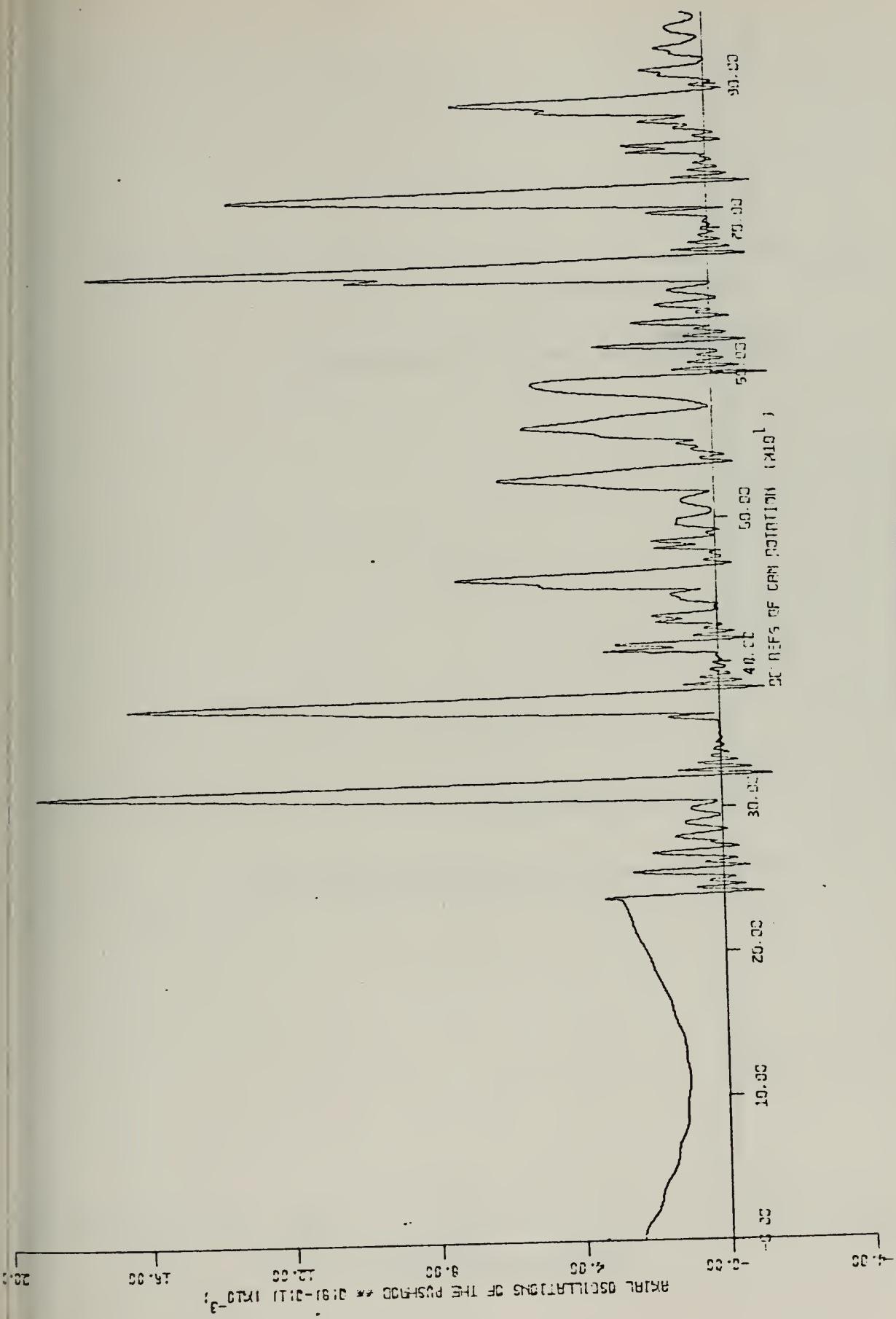
DEGREES OF CRANK ROTATION (deg.) ;
10.00 15.00 20.00 25.00 30.00 35.00 40.00

DEGREES OF CRANK ROTATION (deg.) ;
10.00 15.00 20.00 25.00 30.00 35.00 40.00



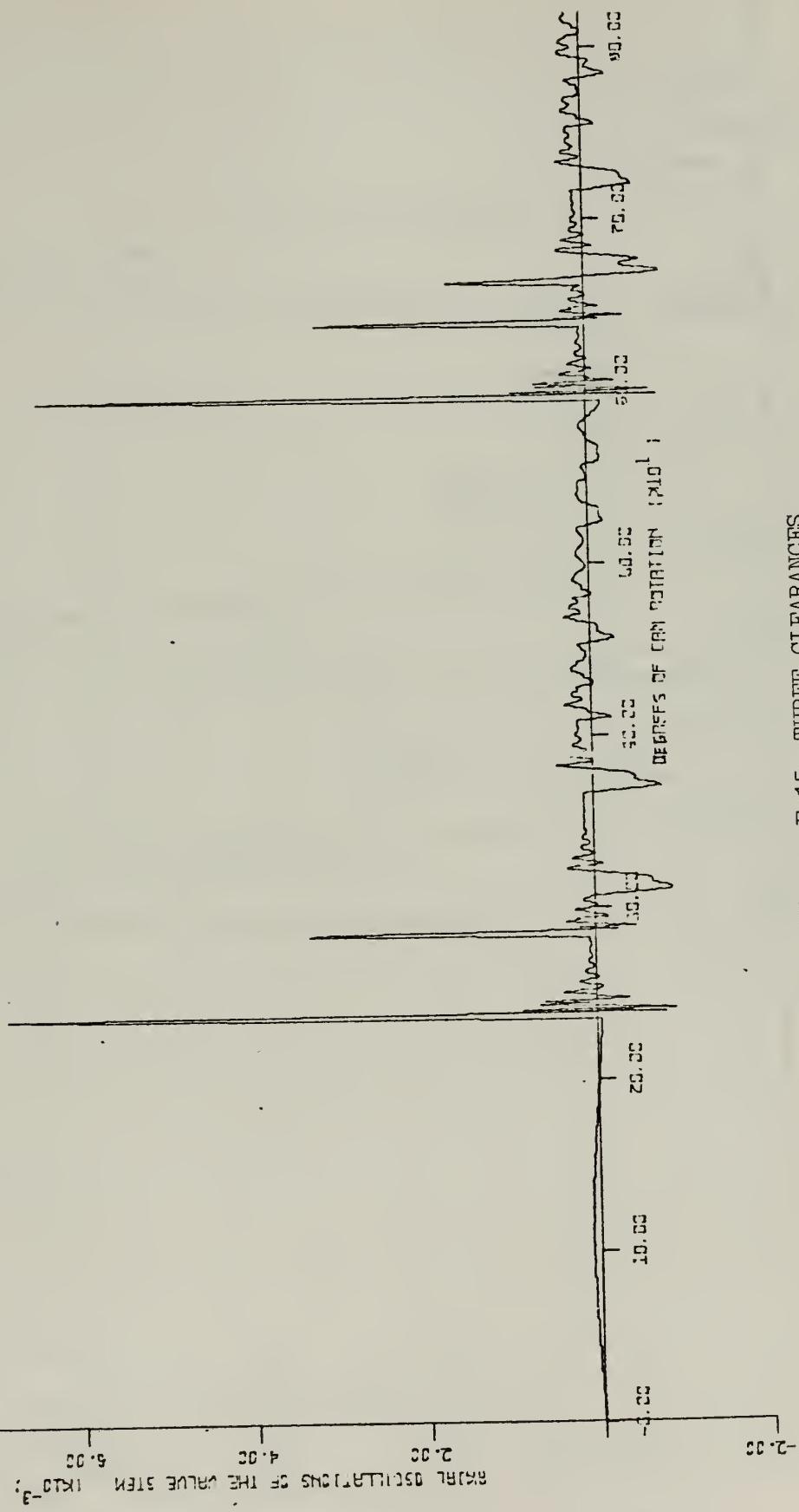
10,000 RPM VALUE SEAT

Figure F.14 THREE CLEARANCES



10,000 RPM VALUE SEAT

Figure F.15 THREE CLEARANCES



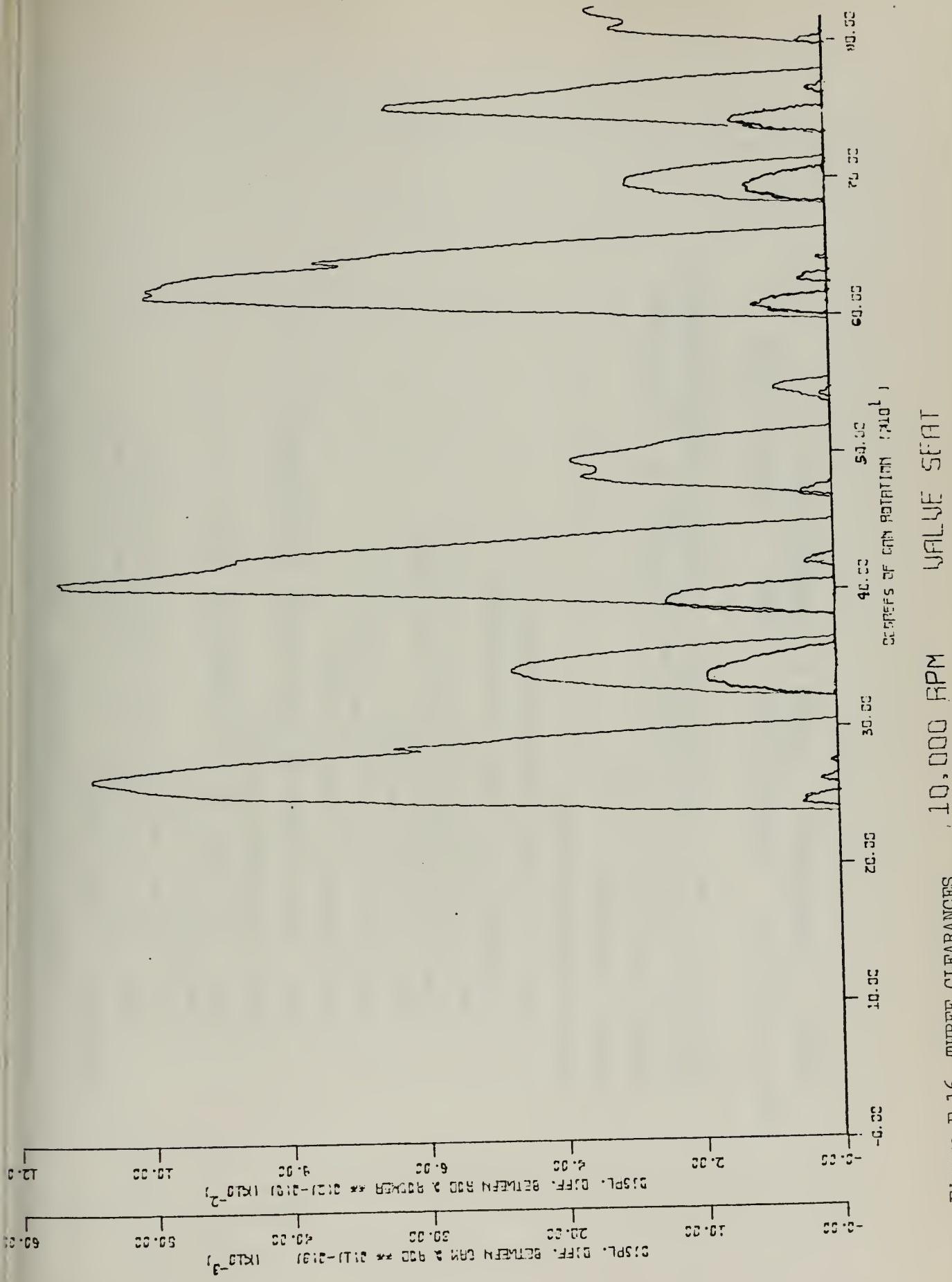


Figure F.16 THREE CLEARANCES

GNILKA, C. W. 17 DECEMBER THREE CLEARANCES NO. 1

11,000 RPM VALVE SEAT

DISPL. DIFF. BETWEEN ROD & ROCKER ** Q(2)-Q(8)

DISPL. DIFF. BETWEEN CAM & ROD ** Q(1)-Q(9)

CAM PROFILE ** Q(9)

DISPL. OF LOWER END OF PUSHROD ** Q(1)

FORCE ON PUSHROD ** F(1)

AXIAL OSCILLATIONS OF THE PUSHROD ** Q(8)-Q(1)

DISPLACEMENT OF VALVE ** Q(7)

POSITION OF VALVE SEAT ** Q(10)

FORCE ON ROCKER ** F(2)

AXIAL OSCILLATIONS OF THE VALVE STEM

FORCE ON VALVE ** F(7)

CAM PROFILE AND DISPL. OF PUSHROD

DISPL. OF VALVE AND VALVE SEAT

NO AGT OUTPUT REQUESTED

DAMPING REQUESTED

CALCOMP GRAPH NOT DESIRED

VALVE SEAT IMPOSED WITH AN INITIAL CLEARANCE OF 0.70 INCHES
COORDINATES 2 AND 8 ARE NOT PINED -- THREE CLEARANCE PROBLEM

RPM = 11000.00 DAMPING COEF. = 1.000 CLEAR = 0.700

XZERO= 8.8245022D-01 FZERO = 5.0185073D 02

IBM-360 COMPUTER PROGRAM LISTING

This listing is coded in FORTRAN IV, (Refs. 10-13), to run on an IBM-360/67, with OS/360 release 18. The simulation requires 234 K bytes of core storage for 850 degrees of cam rotation, including the CALCOMP graph buffer. The offline printer array (GRID) is not included in this core storage figure. Execution time is a function of the speed of rotation of the cam. For 11,000 RPM with 850 degrees, execution time is about 4.3 minutes. The primary output parameters fill arrays for SYSOUT = B and SYSOUT = C. The B queue is for transferring data, via punched cards, for use in the SDS-9300 computer. The C queue is for plotting graphs via the CALCOMP.

CHARLES W. GNILKA - - THESIS - - DECEMBER 1971
THE MODELING AND ANALYSIS OF AN ELASTIC MECHANISM WITH CLEARANCES
THREE CLEARANCES - - CAM - VALVE SEAT - ROCKER

```

1 INTEGER*2 GRID
2 DIMENSION FPLOT(1,1), Y1(1,1), Y2(1,1), Y3(1,1),
3 Y4(1,1), DEG1(1,1), DEG2(1,1), DEG3(1,1),
4 YNAME(20,13), YNAME1(20,2), YNAME2(20,2),
5 COMMON NUNIT EQUIVALENCE (FPLOT(1,1),Y1(1,1),Y2(1,1),Y3(1,1),
6 DEG1(1,1),DEG2(1,1),DEG3(1,1),YNAME(1,1),YNAME1(1,1),
7 YNAME(1,2),YNAME2(1,1),YNAME(1,3),YNAME(1,4),YNAME(1,5),
8 YNAME(1,6),YNAME(1,7),YNAME(1,8),YNAME(1,9),YNAME(1,10),
9 YNAME(1,11),YNAME(1,12),YNAME(1,13),YNAME(1,14),YNAME(1,15),
10 YNAME(1,16),YNAME(1,17),YNAME(1,18),YNAME(1,19),YNAME(1,20),
11 YNAME(2,1),YNAME(2,2),YNAME(2,3),YNAME(2,4),YNAME(2,5),
12 YNAME(2,6),YNAME(2,7),YNAME(2,8),YNAME(2,9),YNAME(2,10),
13 YNAME(2,11),YNAME(2,12),YNAME(2,13),YNAME(2,14),YNAME(2,15),
14 YNAME(2,16),YNAME(2,17),YNAME(2,18),YNAME(2,19),YNAME(2,20),
15 DEG(1,1),DEG(1,2),DEG(1,3),DEG(1,4),DEG(1,5),DEG(1,6),
16 DEG(1,7),DEG(1,8),DEG(1,9),DEG(1,10),DEG(1,11),DEG(1,12),
17 DEG(1,13),DEG(1,14),DEG(1,15),DEG(1,16),DEG(1,17),DEG(1,18),
18 DEG(1,19),DEG(1,20),
19 DATA YNAME(1,1) '/',
20 DATA YNAME(1,2) '/',
21 DATA YNAME(1,3) '/',
22 DATA YNAME(1,4) '/',
23 DATA YNAME(1,5) '/',
24 DATA YNAME(1,6) '/',
25 DATA YNAME(1,7) '/',
26 DATA YNAME(1,8) '/',
27 DATA YNAME(1,9) '/',
28 DATA YNAME(1,10) '/',
29 DATA YNAME(1,11) '/',
30 DATA YNAME(1,12) '/',
31 DATA YNAME(1,13) '/',
32 DATA YNAME(1,14) '/',
33 DATA YNAME(1,15) '/',
34 DATA YNAME(1,16) '/',
35 DATA YNAME(1,17) '/',
36 DATA YNAME(1,18) '/',
37 DATA YNAME(1,19) '/',
38 DATA YNAME(1,20) '/',
39 IP = 0
40 LINES = 80*IP + 1
41 NTIME = 850
42 IGU = YES
43 IGU = NO
44 SCAL = 1.0D0
45 CALL THESIS(FPLOT', DEG, DEG1, DEG2, NTIME, LINES, IP, GRID,
46 Y2, Y3, Y4, YNAME1, YNAME2, SCAL, Y1,
47 STOP
48 END

```



```

2      SUBROUTINE AGT(Q, QR4, WT, WTR4, WI, UNIT, KAGT, XZERO, ROCKER, NPRINT, AGTLTH,
C      IMPLICIT REAL*8(A-H,O-Z)
C      REAL*4 QR4, WTR4, RXZERO, RAMP, RRPM, DIFF
C      INTEGER YES, C
C      DIMENSION Q(1), QR4(1), AGTLTH(1), REALTH(1)
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES/, NO/4HN0 /
C      5001 FORMAT(5E15.7, 13)
C      5002 FORMAT(5E15.7)
C      **** OUTPUT PACKAGE FOR AGT
C      **** CONVERT TO REAL * 4
C      ****
C      IF(KAGT .NE. 1) GO TO 10
C      S = REALTH(2) / AGTLTH(2)
C      THETA = DARSIN(XZERO/REALTH(2))
C      DIFF = DABS((Q(7)-Q(10))/5001) DIFF
C      WRITE(IUNIT,5001) DIFF
C
C      10 QR4(1) = Q(1) - XZERO
C      QR4(2) = Q(2) - XZERO
C      QR4(3) = Q(3) - XZERO
C      QR4(4) = Q(4) + XZERO*ROCKER
C      QR4(5) = Q(5) + THETA}*S
C      QR4(6) = Q(6) + THETA}*S
C      WTR4 = WT
C
C      WRITE(IUNIT,5001) (QR4(JA), JA = 1,5)
C      WRITE(IUNIT,5002) (QR4(JA), JA = 6,9), OMEGA(I), CONFIG(I)
C
C      RETURN
C      END

```



```

C      SUBROUTINE ARRAY(F, Q, QD, QDD, NDEG, FPLOT, NTIME)
C      IMPLICIT REAL*8(A-H,O-Z)
C      REAL*4 FPLOT
C      INTEGER YES
C      DIMENSION F(1), Q(1), QD(1), QDD(1), FPLOT(NTIME,1)
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES/, NO/4HNNO/
C
C      FPLOT(NDEG,1) = Q(2) - Q(8)
C      FPLOT(NDEG,2) = Q(1) - Q(9)
C      FPLOT(NDEG,3) = Q(9) - Q(1)
C      FPLOT(NDEG,4) = Q(1) - Q(1)
C      FPLOT(NDEG,5) = Q(1) - Q(1)
C      FPLOT(NDEG,6) = Q(1) - Q(7)
C      FPLOT(NDEG,7) = Q(7) - Q(10)
C      FPLOT(NDEG,8) = Q(10) - Q(2)
C      FPLOT(NDEG,9) = Q(2) - Q(7)
C      FPLOT(NDEG,10) = Q(7) + Q(4)
C      FPLOT(NDEG,11) = F(7)
C
C      RETURN
C      END

```



```

      2 FORMAT( 'DISPL DIFF. AT COORD 7 = ', 1PD15:7)
      2 DEGREES= , F9:3, CSAVE= , I2:7, C= :I2,
      2 F9:3, 1P3D11:3, FORCES 1:2, 7 = 1, 3D11:3)
      1 FORMAT( 'DISPL 1:2, 7 = ', 1P3D11:3)
      1, 10X, VALUES CALCULATED AFTER BACKSTEP TAKEN: 1)

C   EPS1 = 1.0D-1
C   EPS2 = 1.0D-6

C   PI = 3.14159265359000
C   DEGDP = 180.0D0/PI

C   IK = 0
C   DTPMW = DTPMW*W
C   DELWT = DTPMW/ 500.0D0
C   WT = WT + DTPMW - DTW
C   WTSAY = WT
C   DELDT = DELWT/W

C   IF(NPRINT .EQ. YES) 1 WRITE(6,5007), DTPM, DTPMW, WT, DELWT, DELDT
C   *****
C   *** SAVE OLD PARAMETERS
C   ***
C   *****
C   1 5 DO 5 JA = 1, KSYS
C   5 FP(JA) = FS A V(JA)
C   5 CONTINUE

C   DO 15 JA = 1, NSYS
C   CALL ADJUST(JA, JA, QP, QDP, QDDP, QSAV, QDDSAV)
C   15 CONTINUE

C   IF(NPRINT .EQ. NC) GO TO 20
C   WRITE(6,5003) (FS AV(JA), JA = 1, KSYS)
C   WRITE(6,5004) (QSAV(JA), JA = 1, NSYS)
C   WRITE(6,5005) (QDS AV(JA), JA = 1, NSYS)
C   WRITE(6,5006) (QDDSAV(JA), JA = 1, NSYS)

C   20 IF(NPRINT .EQ. YES) WRITE(NUNIT,5000) DTPM, WT
C   IK = IK + 1

C   CALL CAM(WT, AMP, AMPW, XZERO, W2, QP, QDP, QDDP)
C   1 IF(NPRINT .EQ. YES) 1 WRITE(6,5008), QP(9), QDDP(9),

```



```

C CALL DEL( DTPM, DT02, DT203, DT206, BMIV, DA, DV, SYSM( 1,1,C),
C DAMP(1,1,C), STF(1,i,C), ISYS, C, NPRINT, KSYS, KPRINT,
C
C IF(NPRINT.EQ.YES)
1 WRITE(6,5009) DTPM, C
C CALL BRANCH(C,FP, KSYS,ISYS, QDP,QDDP, QDP, ICORD, DT,
1
C
C IF(NPRINT.EQ.YES)
1 WRITE(6,5010) C, (FP(JA), JA= 1,KSYS)
C CALL NEWMRK( ISYS, KSYS, C, STF, DAMP, SYSM, DP, DTPM, DT02,
C DT206, DT203, QP, QDP, QDDP, BMIV, DA, DV, NC, ALFA,
1
C
C IF(NPRINT.EQ.YES)
1 WRITE(6,5011) (FP(JA), JA= 1,KSYS)
C
C DISPL1 = QP(1) - QP(9)
C DISPL2 = QP(2) - QP(8)
C DISPL7 = QP(7) - QP(10)
C
C IF(NPRINT.EQ.YES)
1 WRITE(6,5013) DISPL1, DISPL2, DISPL7
C
C INDEX = ICORD(IDT,1)
C NDEX = ICORD(IDT,2)
C IF(CONFIG(C, IDT).EQ. 0) GO TO 25
C
C IF(FP( INDEX) .GE. 0.0 ) GO TO 24
C GO TO 30
C
C 24 IF(KPRINT.EQ.YES) AND
C     WRITE(NUNIT,5001) IK, INDX, NDEX, C
C     GO TO 40
C

```



```

00001350
00001360
00001370
00001380
00001390
00001400
00001410
00001420
00001430
00001440
00001450
00001460
00001470
00001480
00001490
00001500
00001510
00001520
00001530
00001540
00001550
00001560
00001570
00001580
00001590
00001600
00001610
00001620
00001630
00001640
00001650
00001660
00001670
00001680
00001690
00001700

25 DISPL = QP(INDEX) - QP(NDEX)
C IF(DISPL .GE. 0) .LE. EPS2) GO TO 28
C GO TO 30
C 28 IF(KPRINT EQ. YES AND
C      WRITE(NUNIT,5012) IK, INDEX, NDEX, C
C      GO TO 40
C
C 30 DTPM = DTPM - DELDT
C      WT = WT - DELWT
C      IF(WT .LE. WTSV) RETURN
C      GO TO 4
C 40 CONTINUE
C      DEGR = WT*DEGOPI
C      NDEG = DEGR
C      WRITE(NUNIT,5014) DEGR, C, ICHNG, DISPL1,DISPL2,DISPL7,
C      1 IF(NPRINT .EQ. NO) RETURN
C      WRITE(NUNIT,5017)
C      CALL DEBUG( C, Q, QD, QDD, F, KSYS, ISYS, DP, NDEG, NPRINT, DEGR,
C      1
C      RETURN
C      END

```



```

000000010
000000020
000000030
000000040
000000050
000000060
000000070
000000080
000000090
000000100
000000110
000000120
000000130
000000140
000000150
000000160
000000170
000000180
000000190
000000200
000000210
000000220
000000230
000000240
000000250
000000260
000000270
000000280
000000290
000000300
000000310
000000320
000000330
000000340
000000350
000000360
000000370
000000380
000000390
000000400
000000410
000000420
000000430
000000440
000000450
000000460

SUBROUTINE BRANCH(C,F,KSYS,ISYS,Q,QD,QDD,QDSAV,ICORD,DT,
1      IMP,PIN2$8, IDIM)
C      IMPLICIT REAL*8(A-H,O-Z)
C      INTEGER YES, C, PIN2$8
C      DIMENSION Q(1), QD(1), QDD(1), QDSAV(1), F(1), ICORD(NCL EAR,1),
1      IDIM(1)

C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES/, NO/4HN0/
C      ISYS = IDIM(C)
C      GO TO ( 10, 20, 30, 40, 50, 60, 70, 80), C
C      CONFIGURATION - I - NO CONTACT
C      F(1) = 0.0DO
C      F(2) = 0.0DO
C      F(7) = 0.0DO
C      GO TO 100

C      CONFIGURATION - II - CONTACT AT VALVE SEAT
C      CONTINUE
C      CALL ADJUST(7,10, Q, QD, QDD, Q, QD, QDD)
C      F(1) = 0.0DO
C      F(2) = 0.0DO
C      GO TO 100

C      CONFIGURATION - III - CONTACT AT ROCKER
C      F(1) = 0.0DO
C      F(7) = 0.0DO
C      GO TO 100

C      CONFIGURATION - IV - CONTACT AT ROCKER AND VALVE SEAT
C      CONTINUE
C      CALL ADJUST(7,10, Q, QD, QDD, Q, QD, QDD)
C      F(1) = 0.0DO

```


GO TO 100

000000479
000000480
000000490
000000510
000000520
000000530
000000540
000000550
000000560
000000570
000000580
000000590
000000600
000000610
000000620
000000630
000000640
000000650
000000660
000000670
000000680
000000690
000000700
000000710
000000720
000000730
000000740
000000750
000000760
000000770
000000780
000000790
000000800
000000810
000000820
000000830
000000840
000000850
000000860
000000870
000000880
000000890
000000900
000000910
000000920
000000930

CONFIGURATION - V - CONTACT AT CAM
CONTINUE
CALL ADJUST(1,9, Q, QD, QDD, Q, QD, QDD)
F(2) = 0.000
F(7) = 0.000
GO TO 100

CONFIGURATION - VI - CONTACT AT CAM AND VALVE SEAT
CONTINUE
CALL ADJUST(1,9, Q, QD, QDD, Q, QD, QDD)
CALL ADJUST(7,10, Q, QD, QDD, Q, QD, QDD)
F(2) = 0.000
GO TO 100

CONFIGURATION - VII - CONTACT AT ROCKER, CAM, AND VALVE SEAT
CONTINUE
CALL ADJUST(1,9, Q, QD, QDD, Q, QD, QDD)
F(7) = 0.000
GO TO 100

CONFIGURATION - VIII - CONTACT AT ROCKER, CAM, AND VALVE SEAT
CONTINUE
CALL ADJUST(1,9, Q, QD, QDD, Q, QD, QDD)
CALL ADJUST(7,10, Q, QD, QDD, Q, QD, QDD)
IF(IMP .NE. 0) QDD(IMP) = (QD(IMP)-QDSAV(IMP)) / DT
RETURN
END


```

SUBROUTINE CONMAT( NC, CONFIG, IDP, NPRINT, KSYS )
C IMPLICIT REAL*8(A-H,O-Z)
C INTEGER CONFIG(16,4), CSAVE, TWON, TWONP1, YES
C DIMENSION IDP(KSYS,1)
C COMMON NUNIT, IPRINT
C DATA YES/4HYES/, NO/4HN0/
C
C *****
C **** ESTABLISH CONFIGURATION
C **** MATRIX
C *****
C 100 NC = 4
C     TWON = 2***NC
C     TWONP1 = TWON + 1
C     NCM1 = NC - 1
C     NCM2 = NC - 2
C     NCM3 = NC - 3
C
C     DO 110 I = 2,TWON,2
C     CONFIG(I,NC) = 1
C 110 CONTINUE
C
C     DO 120 I = 2,TWON,4
C     CONFIG(I+1,NCM1) = 1
C     CONFIG(I+2,NCM1) = 1
C 120 CONTINUE
C
C     DO 130 I = 1,NC
C     CONFIG(I+4,NCM2) = 1
C     CONFIG(I+12,NCM2) = 1
C 130 CONTINUE

```



```

000000410
000000420
000000430
000000440
000000450
000000460
000000470
000000480
000000490
000000500
000000510
000000520
000000530
000000540
000000550
000000560
000000570
000000580
000000590
000000600

***** ESTABLISH ****
*** MATRIX IDP ***
*** ***** ***** *****

DO 140 JA = 1, KSYS
C      IDP(JA,1) = CONFIG(JA,2)
C      IDP(JA,2) = 0
C      IDP(JA,3) = CONFIG(JA,4) * 7
C 140 CONTINUE
C      NC = 8
C      RETURN
END

```



```

1      SUBROUTINE DEBUG(IBLOCK, QD, QDD, F, KSYS, ISYS, DP, NDEC,
C
C      IMPLICIT REAL*8(A-H,O-Z)
C
C      INTEGER YES, C
C
C      DIMENSION Q(10), QD(10), QDD(10), F(10), DP(8)
C
C      COMMON NUNIT, IPRINT
C
C      DATA YES/4HYES/, NO/4HN0/
C
C      5001 FORMAT(' ',*,*****,' ',*,*****,' ',*,*****,' ',*,*****,' ')
C      1      *****,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C
C      5002 FORMAT(' ',*,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C      2      ,*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',
C      3      ,*'D15.7',*'D15.7',*'D15.7',*'D15.7',*'D15.7'
C
C      5003 FORMAT(' ',*,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C      4      ,*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',
C      5      ,*'D15.7',*'D15.7',*'D15.7',*'D15.7',*'D15.7'
C
C      5004 FORMAT(' ',*,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C      6      ,*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',
C      7      ,*'D15.7',*'D15.7',*'D15.7',*'D15.7',*'D15.7'
C
C      5005 FORMAT(' ',*,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C      8      ,*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',
C      9      ,*'D15.7',*'D15.7',*'D15.7',*'D15.7',*'D15.7'
C
C      5006 FORMAT(' ',*,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C      10     ,*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',
C      11     ,*'D15.7',*'D15.7',*'D15.7',*'D15.7',*'D15.7'
C
C      5007 FORMAT(' ',*,*'Q',*'QD',*'QDD',*'DEL P',*'DEL',*' '
C      12     ,*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',*'IP7D16.7',
C      13     ,*'D15.7',*'D15.7',*'D15.7',*'D15.7',*'D15.7'
C
C      5037 FORMAT(' ',*'FORCE',*'16X',*'15',*'DEGREE',*'15.7',*'F9.3',
C      5061 FORMAT(' ',*'CAMQ=1,1P7D16.7 D15.7',*'CONFIGURATION',*'12),
C      5063 FORMAT(' ',*'CAMQ=1,1P7D16.7,11X,*,*'CAMQD = , Di6.8,10X,*,*'CAMQDD = ,',
C      1      Di6.8)
C
C      DISPL1 = Q(1) - Q(9)
C      DISPL2 = Q(2) - Q(8)
C      DISPL7 = Q(7) - Q(10)
C      GO TO(1,2,3,4,5), IBLOCK
C
C      1      CONTINUE
C      WRITE(NUNIT,5061) NDEC,DISPL1,DISPL2,DISPL7
C      WRITE(NUNIT,5007) Q(9),QD(9),QDD(9)
C      WRITE(NUNIT,5063) ( Q(JA),JA = 1, KSYS )
C      WRITE(NUNIT,5002) ( Q(JA),JA = 1, KSYS )
C      WRITE(NUNIT,5003) ( QD(JA),JA = 1, KSYS )
C      WRITE(NUNIT,5004) ( QDD(JA),JA = 1, KSYS )
C      WRITE(NUNIT,5037) ( F(JA),JA = 1, KSYS )
C      IF(NPRINT .EQ. YES) WRITE(NUNIT,5001)
C
C      RETURN
C      C
C      C

```



```

00000470
00000480
00000490
00000500
00000510
00000520
00000530
00000540
00000550
00000560
00000570
00000580
00000590
00000600
00000610
00000620
00000630
00000640
00000650
00000660
00000670
00000680
00000690
00000700
00000710
00000720
00000730

2  CONTINUE
   IF(NDEG .GT. 200)  NPRINT = YES
   WRITE(NUUNIT,5061)  NDEG, DEGR, C2, DISPL7
   WRITE(NUUNIT,5007)  DISPL1, DISPL2, QDD(9)
   WRITE(NUUNIT,5063)  Q(9), QD(9), JA = 1, KSYS
   WRITE(NUUNIT,5002)  Q(JA), JA = 1, KSYS
   WRITE(NUUNIT,5003)  QD(JA), JA = 1, KSYS
   WRITE(NUUNIT,5004)  QDD(JA), JA = 1, KSYS
   WRITE(NUUNIT,5037)  F(JA), JA = 1, KSYS

   RETURN
C   C 3  CONTINUE
C   C 4  CONTINUE
C   C 5  CONTINUE
C   C

```



```

SUBROUTINE DEL( DT, DT02, KPRINT, KSYS, KPRINT, BMIV, DA, DV, EM,
1 SE, CA, KS, C, NPRINT, KSYS, DT203, DT206, BMIV, DA, DV, EM,
C IMPLICIT REAL*8(A-H,O-Z)
C INTEGER YES, C
C DIMENSION BMIV( KSYS,KSYS), DA( KSYS,KSYS), DV( KSYS,KSYS)
C DIMENSION EM( KSYS,KSYS), SE( KSYS,KSYS), CA( KSYS,KSYS)
C COMMON NUNIT, IPRINT
DATA YES/4HYES/, NO/4HN0 /
C 1000 FORMAT('0',5X,'DT =',1PD12.3,', DT**2/6 =',D12.3,
1 DT**2/3 =',D12.3,', DT**2/6 =',D12.3,
C DT02 = DT*0.5D0
C DT203 = DT**2/3*0.D0
C DT206 = DT203*0.5D0
C IF(NPRINT.EQ. NO) GO TO 10
WRITE(NUNIT,1000) DT, DT02, DT203, DT206
C 10 DO 20 JA = 1, KS
C DO 20 BMIV(JA,JB) = 1, KS
C BMIV(JA,JB) = EN(JA,JB) + DT02*SE(JA,JB) + DT206*CA(JA,JB)
C 20 CONTINUE
C CALL DINVS( BMIV, KS,KSYS)
C DO 30 JA = 1, KS
C DO 30 JB = 1, KS
C DA( JA, JB) = 0.0D0
C DV( JA, JB) = 0.0D0
C DO 30 JC = 1, KS
C DA( JA, JB) = DA( JA, JB) + BMIV( JA, JB) + BMIV( JA, JB)
C DV( JA, JB) = DV( JA, JB) + BMIV( JA, JB)
C 30 CONTINUE
C
RETURN
END

```


SUBROUTINE DINVS CODED BY DR. R. C. WINFREY
MODIFIED BY C. W. GNIŁKA

```
C SUBROUTINE DINVS(A,NMAX,KSYS)
C IMPLICIT REAL*8(A-H,O-Z)
C INTEGER YES
C DIMENSION A(KSYS,KSYS)
C COMMON NUNIT, IPRINT
C DATA YES/4HYES /, NO/4HNNO /
C DO 200 N=1, NMAX
D=A(N,N)
C IF(D.EQ.0.) GO TO 300
C DO 100 J=1, NMAX
A(N,J)=-A(N,J)/D
C DO 150 I=1, NMAX
IF(N-1) 110, 150, 110
C DO 140 J=1, NMAX
IF(N-J) 120, 140, 120
C 120 A(I,J)=A(I,J)+A(I,N)*A(N,J)
C 140 CONTINUE
C 150 A(I,N)=A(I,N)/D
A(N,N)=1.0/D
C 200 CONTINUE
C RETURN
N,D
300 WRITE(6,320) N,D, 'ERROR RETURN FROM DINVS. MATRIX SINGULAR. ', 
320 FORMAT('0.', '/10X', 'ELEMENT', 'I4,3H = ,1PD20.12)
1 ,DIAGONAL ELEMENT, I4,3H = ,1PD20.12)
C RETURN
END
```


C C C C C THE FOLLOWING PLOT SUBROUTINES CODED BY DR. R. C. WINFREY

```
00000010  
00000020  
00000030  
00000040  
00000050  
00000060  
00000070  
  
SUBROUTINE DRUP(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)  
DIMENSION TITLE(1),XNAME(1),YNAME(20,1),LABEL(1)  
DIMENSION X(1),Y(N,1)  
CALL DRUBL(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)  
C  
RETURN  
END
```

```
00000080  
00000090  
00000100  
00000110  
00000120  
00000130  
00000140  
00000150  
00000160  
00000170  
00000180  
00000190  
  
SUBROUTINE DRUPB(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)  
DIMENSION TITLE(1),XNAME(1),YNAME(20,1),LABEL(1)  
DIMENSION X(1),Y(N,1)  
CALL DRUB(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)  
C  
RETURN  
END
```

```
00000200  
00000210  
00000220  
00000230  
00000240  
00000250  
00000260  
00000270  
00000280  
00000290  
00000300  
00000310  
  
SUBROUTINE DRUBL(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)  
DIMENSION TITLE(1),XNAME(1),YNAME(20,1),LABEL(1)  
DIMENSION X(1),Y(N,1),YMIN(20),DY(20)  
PAPER=1.  
CALL PLOTS  
CALL PLOT(-11.,0.,-3)  
CALL SYMBOL(0.,0.,14,LABEL,0.,,80)  
CALL PLOT(0.,4.,-3)  
NSIGN=1  
IF(NPT.LT.0) NSIGN=-1  
NPA=NPT*NSIGN
```



```

XST=FLOAT(K)*.65
XSTP=XST+(PAPER-XSIZE)*.5
IF(XSIZE*GT*.01) XSTP=XST+1.
XAXIS=XSIZE-XST
CALL SCALE(X,N,1,XAXIS,0.,XMIN,DX)
DO 100 JA=1,K
CALL SCALE(Y(1,JA),N,1,YSIZE,0.,YMIN(JA),DY(JA))
CONTINUE
100 DO 130 JA=1,N
130 X(JA)=X(JA)+XSTP
YMAX=YMIN(1)+DY(1)*YSIZE
YS=ABS(YMIN(1))/DY(1)
IF(YMAX*LT.0.) YS=0.
IF(YMIN(1).GT.0.) YS=.0.
IF(XSIZE*GT*.01) GO TO 200
CALL CENT(XNAME,NTT)
NLT=-NTT
CALL EXISA(XSTP,YS,XNAME,NLT,XAXIS,0.,XMIN,DX)
CALL EXISA(XSTP,YS,XNAME,NLT,XAXIS,0.,XMIN,DX)
JB=1
DO 160 JA=1,K
JB=JB+1
1F{JB*EQ.*8) JB=.2
YST=XSTP-.65*FLOAT(JA-1)
CALL CENT(YNAME(1,JA),NTT)
CALL EXISA(YST,0.,YNAME(1,JA),NLT,YSIZE,90.,YMIN(JA),DY(JA))
JC=NPT
IF(NPA.EQ.100) JC=JB*N SIGN
CALL LIN(X,Y(1,JA),N,1,JC)
CALL LIN(X,Y(1,JA),N,1,JC)
CALL LIN(X,Y(1,JA),N,1,JC)
CONTINUE
160 YL=-.8
CALL CENT(TITLE,NTT)
XL=PAPER*.5*FLOAT(NTT/2)*.18
CALL SYMBOL(XL, YL,.21,TITLE,0.,NTT)
YAX=YSIZE+.3.
GO TO 300
200 XFAR=(PAPER+YSIZE)*.5
YS=XFAR-YS
CALL CENT(XNAME,NTT)
NLT=-NTT
CALL EXISA(YS,XSTP,XNAME,NLT,XAXIS,90.,XMIN,DX)
JB=1
DO 260 JA=1,K
JB=JB+1
1F{JB*EQ.*8) JB=.2
YST=XSTP-.65*FLOAT(JA-1)

```



```

CALL CENT(YNAME(1,JA),NTT)
CALL EXISA(XFAR,YST,JA),NTT,YSIZE,180.,YMIN(JA),DY(JA))
DO 230 JC=1,N
      Y(JC,JA)=XFAR-Y(JC,JA)
      JC=NPT
      IF(NPA.EQ.10) JC=JB*NSIGN
      CALL LIND(Y(1,JA),X,N,1,JC)
      CALL LIND(Y(1,JA),X,N,1,JC)
      CALL LIND(Y(1,JA),X,N,1,JC)
      DO 260 JC=1,N
          Y(JC,JA)=XFAR-Y(JC,JA)
          CONTINUE
          XC=XFAR+.8
          XF(YSLT+.01) XL=XL+.55
          CALL CENT(TITLE,NTT)
          YL=XSIZE*.5+1.-FL0AT(NTT/2)*.18
          CALL SYMBOL(XL,YL,.21,TITLE,90.,NTT)
          YAX=XSIZE+.3
          CALL PLOT(0,YAX,-3)
          CALL SYMBOL(0,0.,14,LABEL,0.,80)
          CALL PLOT(1.25,4.,-3)
          CALL PLOT(.25,4.,-3)
          WRITE(6,400) TITLE(J),J=1,20
          400 FORMAT(//20X,75HA PLOT HAS BEEN DRAWN BY SUBROUTINE DRUP CONTAI
1NING THE FOLLOWING TITLE - //20X,20A4,///)
          RESTORE ORIGINAL VALUES OF X AND Y
          C
          DO 420 JA=L,N
              X(JA)=(X(JA)-XSTP)*DX+XMIN
          DO 420 JB=1,K
              Y(JA,JB)=Y(JA,JB)*DY(JB)+YMIN(JB)
          RETURN
          END
          230
          260
          300
          420
          440
          460
          480
          500
          520
          540
          560
          580
          600
          620
          640
          660
          680
          700
          720
          740
          760
          780
          800
          820
          840
          860
          880
          900
          920
          940
          950
          960
          970
          980
          990
          1000
          1010
          1020
          1030
          1040
          1050
          1060
          1070
          1080
          1090
          1100
          1110
          1120
          1130
          1140
          1150
          1160
          1170
          1180
          1190
          1200
          1210
          1220
          1230
          1240
          1250
          1260
          1270
          1280
          1290
          1300
          1310
          1320
          1330
          1340
          1350
          1360
          1370
          1380
          1390
          1400
          1410
          1420
          1430
          1440
          1450
          1460
          1470
          1480
          1490
          1500
          1510
          1520
          1530
          1540
          1550
          1560
          1570
          1580
          1590
          1600
          1610
          1620
          1630
          1640
          1650
          1660
          1670
          1680
          1690
          1700
          1710
          1720
          1730
          1740
          1750
          1760
          1770
          1780
          1790
          1800
          1810
          1820
          1830
          1840
          1850
          1860
          1870
          1880
          1890
          1900
          1910
          1920
          1930
          1940
          1950
          1960
          1970
          1980
          1990
          2000
          2010
          2020
          2030
          2040
          2050
          2060
          2070
          2080
          2090
          2100
          2110
          2120
          2130
          2140
          2150
          2160
          2170
          2180
          2190
          2200
          2210
          2220
          2230
          2240
          2250
          2260
          2270
          2280
          2290
          2300
          2310
          2320
          2330
          2340
          2350
          2360
          2370
          2380
          2390
          2400
          2410
          2420
          2430
          2440
          2450
          2460
          2470
          2480
          2490
          2500
          2510
          2520
          2530
          2540
          2550
          2560
          2570
          2580
          2590
          2600
          2610
          2620
          2630
          2640
          2650
          2660
          2670
          2680
          2690
          2700
          2710
          2720
          2730
          2740
          2750
          2760
          2770
          2780
          2790
          2800
          2810
          2820
          2830
          2840
          2850
          2860
          2870
          2880
          2890
          2900
          2910
          2920
          2930
          2940
          2950
          2960
          2970
          2980
          2990
          3000
          3010
          3020
          3030
          3040
          3050
          3060
          3070
          3080
          3090
          3100
          3110
          3120
          3130
          3140
          3150
          3160
          3170
          3180
          3190
          3200
          3210
          3220
          3230
          3240
          3250
          3260
          3270
          3280
          3290
          3300
          3310
          3320
          3330
          3340
          3350
          3360
          3370
          3380
          3390
          3400
          3410
          3420
          3430
          3440
          3450
          3460
          3470
          3480
          3490
          3500
          3510
          3520
          3530
          3540
          3550
          3560
          3570
          3580
          3590
          3600
          3610
          3620
          3630
          3640
          3650
          3660
          3670
          3680
          3690
          3700
          3710
          3720
          3730
          3740
          3750
          3760
          3770
          3780
          3790
          3800
          3810
          3820
          3830
          3840
          3850
          3860
          3870
          3880
          3890
          3900
          3910
          3920
          3930
          3940
          3950
          3960
          3970
          3980
          3990
          4000
          4010
          4020
          4030
          4040
          4050
          4060
          4070
          4080
          4090
          4100
          4110
          4120
          4130
          4140
          4150
          4160
          4170
          4180
          4190
          4200
          4210
          4220
          4230
          4240
          4250
          4260
          4270
          4280
          4290
          4300
          4310
          4320
          4330
          4340
          4350
          4360
          4370
          4380
          4390
          4400
          4410
          4420
          4430
          4440
          4450
          4460
          4470
          4480
          4490
          4500
          4510
          4520
          4530
          4540
          4550
          4560
          4570
          4580
          4590
          4600
          4610
          4620
          4630
          4640
          4650
          4660
          4670
          4680
          4690
          4700
          4710
          4720
          4730
          4740
          4750
          4760
          4770
          4780
          4790
          4800
          4810
          4820
          4830
          4840
          4850
          4860
          4870
          4880
          4890
          4900
          4910
          4920
          4930
          4940
          4950
          4960
          4970
          4980
          4990
          5000
          5010
          5020
          5030
          5040
          5050
          5060
          5070
          5080
          5090
          5100
          5110
          5120
          5130
          5140
          5150
          5160
          5170
          5180
          5190
          5200
          5210
          5220
          5230
          5240
          5250
          5260
          5270
          5280
          5290
          5300
          5310
          5320
          5330
          5340
          5350
          5360
          5370
          5380
          5390
          5400
          5410
          5420
          5430
          5440
          5450
          5460
          5470
          5480
          5490
          5500
          5510
          5520
          5530
          5540
          5550
          5560
          5570
          5580
          5590
          5600
          5610
          5620
          5630
          5640
          5650
          5660
          5670
          5680
          5690
          5700
          5710
          5720
          5730
          5740
          5750
          5760
          5770
          5780
          5790
          5800
          5810
          5820
          5830
          5840
          5850
          5860
          5870
          5880
          5890
          5900
          5910
          5920
          5930
          5940
          5950
          5960
          5970
          5980
          5990
          6000
          6010
          6020
          6030
          6040
          6050
          6060
          6070
          6080
          6090
          6100
          6110
          6120
          6130
          6140
          6150
          6160
          6170
          6180
          6190
          6200
          6210
          6220
          6230
          6240
          6250
          6260
          6270
          6280
          6290
          6300
          6310
          6320
          6330
          6340
          6350
          6360
          6370
          6380
          6390
          6400
          6410
          6420
          6430
          6440
          6450
          6460
          6470
          6480
          6490
          6500
          6510
          6520
          6530
          6540
          6550
          6560
          6570
          6580
          6590
          6600
          6610
          6620
          6630
          6640
          6650
          6660
          6670
          6680
          6690
          6700
          6710
          6720
          6730
          6740
          6750
          6760
          6770
          6780
          6790
          6800
          6810
          6820
          6830
          6840
          6850
          6860
          6870
          6880
          6890
          6900
          6910
          6920
          6930
          6940
          6950
          6960
          6970
          6980
          6990
          7000
          7010
          7020
          7030
          7040
          7050
          7060
          7070
          7080
          7090
          7091
          7092
          7093
          7094
          7095
          7096
          7097
          7098
          7099
          7100
          7101
          7102
          7103
          7104
          7105
          7106
          7107
          7108
          7109
          7110
          7111
          7112
          7113
          7114
          7115
          7116
          7117
          7118
          7119
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          711I
          711J
          711K
          711L
          711M
          711N
          711O
          711P
          711Q
          711R
          711S
          711T
          711U
          711V
          711W
          711X
          711Y
          711Z
          711A
          711B
          711C
          711D
          711E
          711F
          711G
          711H
          7
```


C DRAW AXIS WITH TIC MARKS

XA=X-.06*ZING*STH

YA=Y+.06*ZING*CTH

CALL PLOT(XA,YA,3)

DO 20 I=1,N

CALL PLOT(XB,YB,2)

XC=XB+CTH

YC=YB+STH

CALL PLOT(XC,YC,2)

XA=XA+CTH

YA=YA+STH

CALL PLOT(XA,YA,2)

XB=XC

YB=YC

20 NUMBER THE TIC MARKS

IF(DY) 22,31,22

CHAR=ABS(YMIN)

ABSV=ABS(YMIN+DY)

IF(ABSV-CHAR) 5,6,6

5 ABSV=CHAR

POW=0

CALL ANN(ABSV,BBSV,KPOW)

POW=KPCW

ADY=DY*10.**(-POW)

ABSV=YMIN*10.**(-POW)+TN*ADY

XA=XB-(.18*ZING-.07)*STH-1714*CTH

YA=YB+(.18*ZING-.07)*CTH-1714*STH

N=N+1

DO 30 I=1,N

IF(ABS(ABSV)-EPS) 101,101,100

CALL NUMBER(XA,YA,.14,ABSV,THETA,2)

100 ABSV=ABSV-ADY

101 XA=XA-CTH

30 YA=YA-STH

31 TNC=NAC+7

WRITE AXIS LABEL

XA=X+(SIZE*.5-.06*TNC)*CTH-(-.14*ZING*.46)*STH

YA=Y+(SIZE*.5-.06*TNC)*STH+(-.14*ZING*.46)*CTH

CALL SYMBOL(XA,YA,.14,BCD,THETA,NAC)

XA=XA+((TNC-.6)*.12)*CTH

YA=YA+((TNC-.6)*.12)*STH

802 IF(DY) 801,50,801

801 IF(POW) 35,50,35

CALL SYMBOL(XA,YA,.14,7H(X10),THETA,7)

35 YA=YA+.48*CTH-.14*STH

YA=YA+.48*STH+.14*CTH

IF(POW) 40,50,40

CALL NUMBER(XA,YA,.14,POW,THETA,-1)

40

50 RETURN
END

000001850
000001860

000001870
000001880
000001890
000001900
000001910
000001920
000001930
000001940
000001950
000001960
000001970
000001980
000001990
000002000
000002010
000002020
000002030
000002040
000002050
000002060
000002070
000002080
000002090
000002100
000002110
000002120
000002130
000002140
000002150
000002160
000002170
000002180
000002190
000002200
000002210
000002220
000002230
000002240
000002250
000002260
000002270
000002280
000002290
000002300

SUBROUTINE LINDE(X,Y,N,K,L)
DIMENSION X(1),Y(1)
DATA IPUS,ISQ,IM,ISTAR,ICROSS,IDOT,
C /,+/, ZCA000000, ZCB000000, *!, ZCC000000, ./

13=3
KK=K
C...DATA IN EVERY KTH CELL
IF(K)5,55,6
55 WRITE(6,1000)
1000 FORMAT(5X,PLT CODE 01 K=0 IN ARG LIST FOR SUBROUT. LINE*)
RETURN

5 I3=2 KK=NPKK
C...TEST BRANCH VALUE
LL=IABS(L)
IF(LL)65,65,64
64 IF(LL-7)7,7,65
65 LL=1
DO 10 I=1,NP,KK
CALL PLOT(X(I),Y(I),I3)
C...PEN CONDITION BETWEEN POINTS
IF(LL)71,71,73
71 I3=3
73 I3=2 GO TO 199
C..:BRANCH TO SPECIFIED MARKER
199 GO TO (10,200,300,400,500,600,700),LL
200 IBCD=IPUS
GO TO 9
300 IBCD=ISQ
GO TO 91
400 IBCD=IDM
GO TO 9
500 IBCD=ISTAR
GO TO 91
600 IBCD=ICROSS
GO TO 91
700 IBCD=IDOT
XP=X(I)-.03
YP=Y(I)-.03
9 GO TO 94
YP=Y(I)-.06


```

90 GO TO 94
91 XP=X(I)-.03
92 YP=Y(I)-.05
93 GO TO 94
94 CALL SYMBOL(XP,YP,14,I1,3)
95 CALL PLOT(X(I),Y(I),3)
10 CONTINUE
11 RETURN
END

```

```

SUBROUTINE CENT(BCD,NTT)
DIMENSION BCD(1),
DATA BLNK/, /,
NTT=0
DO 10 JA=1,20
JB=JA
IF(BCD(JA).NE.BLNK) GO TO 20
10 CONTINUE
20 JA=20
IF(BCD(JA).NE.BLNK) GO TO 30
21 IF(JA.LT.0) RETURN
GO TO 21
JC=JA
JA=JA-1
IF(JA.LT.0) RETURN
30 NTTA=JC-JB+1
NTT=4*NTTA
DO 50 JA=1,NTTA
JD=JA+JB-1
BCD(JA)=BCD(JD)
50 NTTP=NTTA+1
IF(NTTP.GT.20) RETURN
DO 60 JA=NTTP,20
BCD(JA)=BLNK
60 RETURN
END

```

```

SUBROUTINE DRUA(X,Y,K,N,XSIZE,YSIZE,NPT)
DIMENSION X(1),Y(N,1),YMIN(20),DY(20)
DIMENSION TITLE(20),XNAME(20),YNAME(20,20)
PAPER=11
50 FORMAT(20A4)
READ(5,50,END=400) TITLE(JA),JA=1,20
READ(5,50,END=400) XNAME(JA),JA=1,20
DO 60 JA=1,K
READ(5,50,END=400) YNAME(JB,JA),JB=1,20

```


60

```

CONTINUE
CALL PLOTS
CALL PLOT(-1.1,0.,0.,-3)
CALL PLOT(0.,4.,-3)
NSIGN=1
IF(NPT LT.0) NSIGN=-1
NPA=NPT*T*NSIGN
XST=FLOAT(K)**.5
XSTP=XST+(PAPER-XSIZE)*.5
IF(XSIZE GT.8.01) XSTP=XST+1.
XAXIS=XSIZE-XST
CALL SCALE(XN,1,XAXIS,0.,XMIN,DX)
DO 100 JA=1,K
CALL SCALE(Y(1,JA),N,1,YSIZE,0.,YMIN(JA),DY(JA))
CONTINUE
DO 130 JA=1,N
X(JA)=X(JA)+XS TP
YMAX=YMIN(1)+DY(1)*YSIZE
YS=ABS(YMIN(1))/DY(1)
IF(YMAX LT.0) YS=0
IF(YMIN(1) GT.0) YS=0
IF(XSIZE GT.8.01) GO TO 200
CALL CENT(XNAME,NTT)
NLT=-NTT
CALL EXIS(XSTP,YS,XNAME,NLT,XAXIS,0.,XMIN,DX)
JB=1
DO 160 JA=1,K
JB=JB+1
IF(JB EQ.8) JB=2
YST=XSTP-.5*FLOAT(JA-1)
CALL CENT(YNAME(1,JA),NTT)
CALL EXIS(YST,0.,YNAME(1,JA),NLT,YSIZE,90.,YMIN(JA),DY(JA))
JC=NPT
IF(NPA EQ.10) JC=JB*NSIGN
CALL LINE(X,Y(1,JA),N,1,JC)
CONTINUE
YL=-.8
CALL CENT(TITLE,NTT)
XL=PAPER*.5-FLOAT(NTT/2)*.12
CALL SYMBOL(XL,YL,.14,TITLE,0.,NTT)
CALL EXIS(YFAR-Y,NTT)
GO TO 200
XFAR=(PAPER+YSIZE)*.5
YS=XFAR-YS
CALL CENT(XNAME,NTT)
NLT=-NTT
CALL EXIS(YS,XSTP,XNAME,NLT,XAXIS,90.,XMIN,DX)

```



```

JB=1
DO 260 JA=1,K
JB=JB+1
IF(JB.EQ.8) JB=2
YST=XSTP-.5*FLOAT(JA-1)
CALL CENT(YNAME(1,JA),NTT)
CALL EXIS(XFAR,YST,YNAME(1,JA),NTT,YSIZE,180.,YMIN(JA),DY(JA))
DO 230 JC=1,N
Y(JC,JA)=XFAR-Y(JC,JA)
CONTINUE
XL=XFAR+.8
CALL CEN(TITLE,NTT)
YL=XSIZE*.5+.12
CALL SYMBOL(XL,YL,.14,TITLE,90.,NTT)
YAX=XSIZE+.3
CALL PLOT(0.,YAX,-3)
CALL SYMBOL(0.,0.,.21,'BOX 81 WINFREY',0.,15)
CALL PLOT(1.25,4.,-3)
CALL PLOT(.75,1.75)
WRITE(6,380) TITLE(J,J=1,20)
FORMAT(1H1//20X,75HA PLOT HAS BEEN DRAWN BY SUBROUTINE DRUA CONTAI
N THE FOLLOWING TITLE // X AND Y // /20X,20A4,///)
C RESTORE ORIGINAL VALUES TO X AND Y
DO 390 JA=1,N
X(JA)=(X(JA)-XSTP)*DX+XMIN
DO 390 JB=1,K
Y(JA,JB)=Y(JA,JB)+YMIN(JB)
390 RETURN
400 JA=2+K
WRITE(6,450) JA
450 FORMAT(1H1//20X,43HNOT ENOUGH DATA CARDS FOR SUBROUTINE DRUA.
1 //122,52H CARDS ARE READ EACH TIME SUBROUTINE DRUA IS CALLED.
2 //20X54H RETURNED TO CALLING ROUTINE WITHOUT STARTING A PLOT.
3 /1H1//)
RETURN
END

SUBROUTINE DRUB(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)
DIMENSION X(1),Y(1),YM(20),DY(20)
DIMENSION TITLE(1),XNAME(1),YNAME(1),LABEL(1)
PAPER=11
CALL PLOTS

```



```

CALL PLOT(-11.,0.,-3)
CALL SYMBOL(0.,0.,14,LABEL,0.,80)
CALL PLOT(0.,4.,-3)
NSIGN=1
IF(NPT LT 0) NSIGN=-1
NPA=NPT*K**5
NST=FLOAT(K)**5
XSTP=XST+(PAPER-XSIZE)*.5
IF(XSIZE GT 8.01) XSTP=XST+.1.
XAXIS=XSIZE-XST
CALL SCALE(XN,1,XAXIS,0.,XMIN,DX)
DO 100 JA=1,K
CALL SCALE(Y(1,JA),N,1,YSIZE,0.,YMIN(JA),DY(JA))
CONTINUE
DO 130 JA=1,N
X(JA)=X(JA)+XSTP
YMAX=YMIN(1)+DY(1)*YSIZE
YS=ABS(YMIN(1))/DY(1)
IF(YMAX LT 0.) YS=0.
IF(YMIN(1) GT 0.) YS=0.
IF(XSIZE GT 8.01) GO TO 200
CALL CENT(XNAME,NTT)
NLT=-NTT
CALL EXIS(XSTP,YS,XNAME,NLT,XAXIS,0.,XMIN,DX)
JB=1
DO 160 JA=1,K
JB=JB+1
IF(JB EQ 8) JB=2
YST=XSTP-.50*FLOAT(JA-1)
CALL CENT(YNAME(1,JA),NTT)
CALL EXIS(YST,O.,YNAME(1,JA),NTT,YSIZE,90.,YMIN(JA),DY(JA))
JC=NPT
IF(NPA EQ 10) JC=JB**NSIGN
CALL LINE(X,Y(1,JA),N,1,JC)
CONTINUE
160
Y=~-8
CALL CENT(TITLE,NTT)
XL=PAPER*.5-FLOAT(NTT/2)*.12
CALL SYMBOL(XL,YS,NTT)
YAX=YSIZE+.3.
GO TO 300
XFAR=(PAPER+YSIZE)*.5
YS=XFAR-YS
CALL CENT(XNAME,NTT)
NLT=-NTT
CALL EXIS(YS,XSTP,XNAME,NLT,XAXIS,90.,XMIN,DX)
JB=1
DO 260 JA=1,K

```



```

JB=JB+1
IF(JB.EQ.8) JB=2
YST=XSTP-.50*FLOAT(JA-1)
CALL CENT(YNAME(1,JA),NTT)
CALL EXIS(XFAR,YST,YNAME(1,JA),NTT,YSIZE,180.,YSIZE,180.,YMIN(JA),DY(JA))
DO 230 JC=1,N
Y(JC,JA)=XFAR-Y(JC,JA)
230 JC=NPT
CONTINUE
IF(NPA.EQ.10) JC=JB*N SIGN
CALL LINE(Y(1,JA),X,N,1,JC)
DO 260 JC=1,N
Y(JC,JA)=XFAR-Y(JC,JA)
260 CONTINUE
XL=XFAR+.8
IF(YSLT.01) XL=XL+.55
CALL CENT(TITLE,NTT)
YL=XSIZE*.5+1-FLOAT(NTT/2)*.12
CALL SYMBOL(XL,YL,.14,TITLE,90.,NTT)
YAX=XSIZE+.3
CALL PLOT(0,YAX,-3)
300 CALL SYMPOL(0,0.,.14,LABEL,0.,80)
CALL PLOT(1.25,4.,-3)
CALL PLOT(6,400)(TITLE(J),J=1,20)
WRITE(6,400) (TITLE(J),J=1,20)
FORMAT(//20X,75HA,PLOT HAS BEEN DRAWN BY SUBROUTINE DRUB CONTAI
1NING THE FOLLOWING TITLE //20X,20A4,///)
C RESTORE ORIGINAL VALUES OF X AND Y
DO 420 JA=1,N
X(JA)=(X(JA)-XSTP)*DX+XMIN
DO 420 JB=1,K
420 Y(JA,JB)=Y(JA,JB)*DY(JB)+YMIN(JB)
RETURN
END

```

```

SUBROUTINE DRUC(X,Y,XNAME(1),YNAME(1),NPT,LABEL,TITLE,XNAME,YNAME)
DIMENSION TITLE(1),XNAME,XSIZE,YSIZE,NPT,LABEL(1),X(N,1),Y(N,1)
PAPER=11
CALL PLOTS
CALL PLOT(-11.,0.,-3)
CALL SYMBOL(0.,0.,-3)
CALL PLOT(0.,4.,-3)
NSIGN=1
IF(NPT.LT.0) NSIGN=-1
NPA=NPT*N SIGN
XST=1.
XSTP=XST+(PAPER-XSIZE)*.5
IF(XSIZE.GT.8.01) XSTP=XST+1.

```



```

XAXIS=XSIZE-XST
NK=N*K
CALL SCALE(X,NK,1,XAXIS,0.,XMIN,DY)
DO 130 JA=1,K
  DO 130 JB=1,N
    X(JA,JB)=X(JA,JB)+XSTP
    YMAY=YMINT+DY*YSIZE
    YS=ABS(YMIN/DY)
    IF(YMAX.LT.0.) YS=0.
    IF(YMIN.GT.0.) YS=0.
    IF(YSIZE.GT.*8.01) GO TO 200
    CALL CENT(XNAME,NTT)
    NLT=-NTT
    CALL EXIS(XSTP,YS,XNAME,NLT,XAXIS,0.,XMIN,DY)
    CALL CENT(YNAME,NTT)
    CALL EXIS(XSTP,0.,YNAME,NTT,YSIZE,90.,YMIN,DY)
    JB=1
    DO 160 JA=1,K
      JB=JB+1
      IF(JB.EQ.8) JB=2
      JC=NPT
      IF(NPA.EQ.10) JC=JB*NSIGN
      CALL LINE(X(1,JA),Y(1,JA),N,1,JC)
      CONTINUE
      YL=-.8
      CALL CENT(TITLE,NTT)
      XL=PAPER*.5-FLAT(NTT/2)*T12
      CALL SYMBOL(XL,YL,.14,TITLE,0.,NTT)
      YAX=YSIZE+3.
      GO TO 300
      XFAR=(PAPER+YSIZE)*.5
      YS=XFAR-Y
      CALL CENT(XNAME,NTT)
      NLT=-NTT
      CALL EXIS(YS,XSTP,XNAME,NLT,XAXIS,90.,XMIN,DY)
      CALL CENT(YNAME,NTT)
      CALL EXIS(XFAR,XSTP,YNAME,NTT,YSIZE,180.,YMIN,DY)
      JB=1
      DO 260 JA=1,K
        JB=JB+1
        IF(JB.EQ.8) JB=2
        DO 230 JC=1,N
          Y(JC,JA)=XFAR-Y(JC,JA)
          IF(NPA.EQ.10) JC=JB*NSIGN
          CALL LINE(Y(1,JA),X(1,JA),N,1,JC)
        DO 260 JC=1,N

```



```

260 Y(JC,JA)=XFAR-Y(JC,JA)
CONTINUE
XL=XFAR+.8
CALL CENT(TITLE,NTT)
YL=XSIZE*.5+.1*FLOAT(NTT/2)*.12
CALL SYMBOL(XL,YL,.14,TITLE,90.,NTT)
YAX=XSIZE+.3
CALL PLOT(0,YAX,-3)
CALL SYMBOL(0,0.,14,LABEL,0.,80)
CALL PLOT(1.25,4.,-3)
CALL PLOTE(6,400)(TITLE(J),J=1,20)
FORMAT(//20X,75HA PLOT HAS BEEN DRAWN BY SUBROUTINE DRUC CONTAI
1NING THE FOLLOWING TITLE - AND Y
C RESTORE ORIGINAL VALUES TO X AND Y
DO 420 JA=1,N
DO 420 JB=1,K
X(JA,JB)=(X(JA,JB)-XSTP)*DX+XMIN
Y(JA,JB)=Y(JA,JB)*DY+YMIN
420 RETURN
END

```



```

SUBROUTINE EXIS (X,Y,BCD,NC,SIZE,THETA,YMIN,DY)
DIMENSION BCD(1)
DATA EPS/1.E-6/
ZING=1.*1.2,2
1 ZING=-1.*1
2 NAC=IABS(NC)
TH=THETA*.0174533
N=SIZE+.5
CTH=COS(TH)
STH=SIN(TH)
TN=N
XB=X
YB=Y
XA=X-.1*ZING*STH
YA=Y+.1*ZING*CTH
CALL PLOT(XA,YA,3)
DO 20 I=1,N
CALL PLOT(XB,YB,2)
XC=XB+CTH
YC=YB+STH
CALL PLOT(XC,YC,2)
XA=XA+CTH
YA=YA+STH
CALL PLOT(XA,YA,2)

```



```

XB=XC
YB=YC
20 IF(DY) 22,31,22
CHAR=ABS(YMIN)
IF(ABSV=ABS(YMIN+DY))
5 ABSV=CHAR 5,6,6
POW=0
6 CALL ANN(ABSV,BBSV,KPOW)
POW=KPOW
93 ADY=DY*10.***(-POW)
ABSV=YMIN*10.***(-POW)+TN*ADY
XA=XB-(.2*ZING-.05)*STH-.0857*CTH
YA=YB+(.2*ZING-.05)*CTH-.0857*STH
N=N+1
DO 30 I=1,N
IF(ABSV-EPS) 101,101,100
CALL NUMBER(XA,YA,.1,ABSV,THETA,2)
101 ABSV=ABSV-ADY
XA=XAX-CTH
30 YA=YA-CTH
31 TNC=NACTH
XA=X+(SIZE*.5-.03*TNC)*CTH-(-.07+ZING*.36)*STH
YA=Y+(SIZE*.5-.03*TNC)*STH+(-.07+ZING*.36)*CTH
CALL SYMBOL(XA,YA,.07,BCD,THETA,NAC)
XA=XA+((TNC-.6.)*.06)*CTH
YA=YA+((TNC-.6.)*.06)*STH
IF(DY) 801,50,801
801 IF(POW) 35,50,35
35 CALL SYMBOL(XA,YA,.07,7H(X10 ),THETA,7)
XA=XA+.24*CTH-.07*STH
YA=YA+.24*STH+.07*CTH
IF(POW) 40,50,40
40 CALL NUMBER(XA,YA,.1,POW,THETA,-1)
50 RETURN
END

```

```

SUBROUTINE DRUQ(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)
DIMENSION TITLE(1),XNAME(1),YNAME(1),LABEL(1),X(N1),Y(N1)
CALL DRUCL(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)
RETURN
END

```



```

SUBROUTINE DRUCL(X,Y,K,N,XSIZE,YSIZE,NPT,LABEL,TITLE,XNAME,YNAME)
DIMENSION TITLE(1),XNAME(1),YNAME(1),LABEL(1),X(N,1),Y(N,1)
PAPER=11
CALL PLOTS
CALL PLOT(-11.,0.,-3)
CALL SYMBOL(0.,0.,14,LABEL,0.,80)
CALL PLOT(0.,4.,-3)
NSIGN=N
IF(NPT.LT.0) NSIGN=-1
NPA=NPT*NSIGN
XST=1*XSI+(PAPER-XSIZE)*.5
XIF(XSIZE.GT.8.01) XSTP=XST+1.
NK=N*K
XAXIS=XSIZE-XST
CALL SCALE(X,NK,1,XAXIS,0.,XMIN,DX)
CALL SCALE(Y,NK,1,YSIZE,0.,YMIN,DY)
DO 130 JB=1,K
DO 130 JA=1,N
X(JA,JB)=X(JA,JB)+XSTP
YMAX=YMIN+DY*YSIZE
YS=ABSYMIN/DY
IF(YMAX.LT.0.) YS=0.
IF(YMIN.GT.0.) YS=0.
IF(XSIZE.GT.8.01) GO TO 200
CALL CENT(XNAME,NT)
NLT=-NT
CALL EXISA(XSTP,YS,XNAME,NLT,XAXIS,0.,XMIN,DX)
CALL CENT(YNAME,NT)
CALL EXISA(XSTP,0.,YNAME,NTT,YSIZE,90.,YMIN,DY)
JB=1
DO 160 JA=1,K
JB=JB+1
IF(JB.EQ.8) JB=2
JC=NPT
IF(NPA.EQ.10) JC=JB*NSIGN
CALL LIND(X(1,JA),Y(1,JA),N,1,JC)
CALL LIND(X(1,JA),Y(1,JA),N,1,JC)
CALL LIND(X(1,JA),Y(1,JA),N,1,JC)
CONTINUE
160
YL=-.8
CALL CENT(TITLE,NT)
XL=PAPER*.5-FLOAT(NT/2)*.18
CALL SYMBOL(XL,YL,.21,TITLE,0.,NTT)
YAX=YSIZE+3.
GO TO 300
200
XFAR=(PAPER+YSIZE)*.5
YS=XFAR-Y

```



```

CALL CENT(XNAME,NTT)
NL T=-NTT
CALL EXISA(YS,XSTP,XNAME,NLT,XAXIS,90.,XMIN,DX)
CALL CENT(YNAME,NTT)
CALL EXISA(XFAR,XSTP,YNAME,      NTT,YSIZE,180.,YMIN,DY)

JB=1
DO 260 JA=1,K
JB=JB+1
IF(JB.EQ.8) JB=2
DO 230 JC=1,N
Y(JC,JA)=XFAR-Y(JC,JA)
JC=NPT
IF(NPA.EQ.10) JC=JB*NSIGN
CALL LIN(1,JA),X(1,JA),N,1,JC)
CALL LIN(1,JA),X(1,JA),N,1,JC)
CALL LIN(1,JA),X(1,JA),N,1,JC)
DO 260 JC=1,N
Y(JC,JA)=XFAR-Y(JC,JA)
CONTINUE
XL=XFAR+.8
CALL CENT(TITLE,NTT)
YL=XSIZE*.5+1.-FLOAT(NTT/2)*.18
CALL SYMBOL(XL,YL,.21,TITLE,90.,NTT)
YAX=XSIZE+.3.
300 CALL PLOT(0,YAX,-3)
CALL SYMBOL(0,0,0,14,LABEL,0.,80)
CALL PLOT(.25,4.,-3)
CALL PLOT(.1,25,4.,-3)
WRITE(6,400) (TITLE(J),J=1,20)
400 FORMAT(1//20X,75HA PLOT HAS BEEN DRAWN BY SUBROUTINE DRUQ CONTAI
1NING THE FOLLOWING TITLE - AND Y //20X,20A4,///)
C RESTORE ORIGINAL VALUES TO X AND Y
DO 420 JA=1,N
DO 420 JB=1,K
X(JA,JB)=(X(JA,JB)-XSTP)*DX+XMIN
Y(JA,JB)=Y(JA,JB)*DY+YMIN
RETURN
END

```



```

C SUBROUTINE DTPRIM( RECENT, KK, OLD, DT, DTP, NPRINT )
C IMPLICIT REAL*8(A-H,O-Z)
C INTEGER YES
C DIMENSION DTP(1)
C COMMON NUNIT, IPRINT
C DATA YES/4HYES /, NO/4HNO /
C 5001 FORMAT('      10X, 'FOR INTERATION ', ', RECENT = '
C           'OLD - RECENT = ', D15.7, I1)
C           AND OLD = ', D15.7, I1)
C           SMALL = 1.0D-10
C           DIFF = OLD - RECENT
C           IF(DIFF .LT. SMALL) DIFF = SMALL
C           DTP(KK) = ( OLD/ DIFF ) * DT
C           IF(NPRINT .EQ. NO) RETURN
C           WRITE(NUNIT,5001) KK, RECENT, OLD, DIFF
C           RETURN
C END

```


C SUBROUTINE CODED BY DR. R. C. WINFREY

SUBROUTINE DVVS(A,B,ELAM,X,N,NA,NB,NX,KSYS,NERR)

THIS ROUTINE FINDS THE EIGENVALUES AND ON OPTION THE
EIGENVECTORS OF THE SYSTEM (A-ELAM*B)X=0

A= SYMMETRIC MATRIX--DESTROYED BY ROUTINE
B= POSITIVE DEFINITE MATRIX--DESTROYED BY ROUTINE
ELAM= VECTOR WHERE EIGENVALUES WILL BE STORED
X= MATRIX WHERE EIGENVECTORS WILL BE STORED BY COLUMNS
N= ORDER OF MATRICES A, B AND X AND VECTOR ELAM
NA= ROW DIMENSION OF MATRIX A (GT OR =N)
NB= ROW DIMENSION OF MATRIX B (GT OR =N)--IF B IS IDENTITY
SET NB=0 THEN STORAGE FOR B IS NOT NECESSARY
NX= ROW DIMENSION OF MATRIX X (GT OR =N)--IF VECTORS ARE NOT
DESIRED SET NX=0, THEN STORAGE FOR X IS NOT NECESSARY
NERR= RETURN INDICATOR
=0, NORMAL RETURN
=NEGATIVE, SOLUTION DID NOT CONVERGE
=POSITIVE, B IS NOT POSITIVE DEFINITE AND NERR CONTAINS
THE INDEX OF DIAGONAL ELEMENT CAUSING THIS CONDITION

DOUBLE PRECISION A(KSYS,1),B(KSYS,1),X(KSYS,KSYS),ELAM(KSYS)
DOUBLE PRECISION TBI,TI,T2,T3,TL1,TAI,TBJ,TLS,C,TLI,EMAX
NN=N
NERR=0
NNX=NX

C IF EIGENVECTORS ARE REQUESTED STORE IDENTITY AT X
C IF (NNX.EQ.0)GO TO 20
DO 15 I=1,NN
DO 10 J=1,NN
10 X(I,J)=0.0
X(I,I)=1.0
C IF B=I MATRIX REDUCTION NOT NECESSARY
C 20 IF (NB.EQ.0)GO TO 200

C MATRIX REDUCTION---FIND MATRIX M SUCH THAT $M^T B M = I$
 C THE ACTUAL COMPUTATIONS OF M (UPPER TRIANGULAR) IS ONLY
 C DONE WHEN EIGENVECTORS ARE REQUESTED IT IS STORED AT Y
 C REDUCE B TO I (UPPER TRIANGULAR)
 C REDUCE A TO $A_1 = M^T A M$ (LOWER TRIANGULAR)
 C THE EIGENVALUES OF $(A - ELAM * B) X = 0$ EQUAL THOSE OF $(A_1 - ELAM * I) Y = 0$
 C THE EIGENVECTORS HAVE THE RELATION $X = YM$

```

DO 150 I=1,NN
  TB1=B(I,I)
  IF(TB1.GT.0.0) GO TO 100
  RETURN
  NERR=I
  T1=1.0/TB1
  TAI=A(I,I)
  IM1=I-1
  IF(I.EQ.NN) GO TO 130
  IP1=I+1
  DO 125 J=IP1,NN
    TBJ=B(I,J)
    ELAM(J)=-TBj*T1
    TLJ=ELAM(J)
    IF(1M1.EQ.0) GO TO 110
    DO 105 K=I,IM1
      A(J,K)=TLJ*A(I,K)+A(J,K)
      T2=TLJ*TAI+A(J,I)
      DO 115 K=I,P1,J
        B(K,J)=B(K,J)+ELAM(K)*TBj
        A(J,K)=ELAM(K)*T2+TLJ*A(K,I)+A(J,K)
        IF(NNX.EQ.0) GO TO 125
    C IF NO VECTORS ARE REQUIRED M IS NOT COMPUTED
    C
    DO 120 K=1,I
      X(K,J)=X(K,I)*TLJ+X(K,J)
    125 CONTINUE
    DO 127 J=IP1,NN
      A(J,I)=TLI*(ELAM(J)*TAI+A(J,I))
    127 IF(1M1.EQ.0) GO TO 140
    130 DO 135 K=I,IM1
      A(I,K)=TLI*A(I,K)
    135 IF(NNX.EQ.0) GO TO 150
    140 DO 145 K=I,I
      X(K,I)=X(K,I)*TLI
    145 A(I,I)=T1*TAI
    150
  
```


C SEQUENTIAL THRESHOLD JACOBI METHOD

C 200 NM1=NN-1
C INITIALIZE THRESHOLD--FIND MAXIMUM OFF DIAGONAL ELEMENT
C
C IS=6
C EMAX=0 . 0
C DO 215 J=1,NM1
C JP1=J+1
C DO 215 I=JP1,NN
C T1=DABS(A(I,J))
C IF(T1.GT.EMAX) EMAX=T1
C CONTINUE
C EMAX=EMAX/4 . 0
C
C DIAGONALIZATION SWEEP
C
C 300 IST=0
C DO 350 J=1,NM1
C JP1=J+1
C DO 350 I=JP1,NN
C T1=A(I,J)
C IF(DABS(T1).LE.EMAX) GO TO 350
C IST=IST+1
C
C FORM ROTATION MATRIX ELEMENTS C AND S
C
C T2=A(J,J)-A(I,I)
C IF(T2.EQ.0.0)GO TO 310
C T2=2.0*(T1/T2)
C IF(DABS(T2).GE.1.0)GO TO 310
C T2=T2/(1.0+DSQRT(1.0+T2*T2))
C C=1.0/DSQRT(1.0+T2*T2)
C S=C*T2
C GO TO 320
C C=1./DSQRT(2.0)
C IF(T1.LE.0.0)GO TO 315
C S=C
C GO TO 320
C S=-C
C
C 315 REDUCE A BY ROTATION USING COLUMNS AND ROWS I AND J
C
C 320 IM1=I-1
C T3=C*A(I,J)-S*A(J,J)
C A(J,J)=C*A(J,J)+S*A(I,J)
C IF(I.EQ.JP1) GO TO 327


```

DO 325 K=JP1,IM1
T1=A(K,J)
T2=A(I,K)
A(K,J)=C*T1+S*T2
A(I,K)=C*T2-S*T1
DO 330 K=I,NN
T1=A(K,J)
T2=A(K,I)
A(K,J)=C*T1+S*T2
A(K,I)=-S*T1+C*T2
DO 335 K=1,J
T1=A(J,K)
A(J,K)=C*T1+S*T2
A(I,K)=-S*T1+C*T2
A(I,I)=-S*T3+C*A(I,I)

C FORM EIGENVECTORS IF REQUESTED
C IF (NNX.EQ.0)GO TO 350
DO 345 K=1,NN
T1=X(K,J)
T2=X(K,I)
X(K,J)=C*T1+S*T2
X(K,I)=-S*T1+C*T2
CONTINUE
C TEST FOR CONVERGENCE AND MODIFY THRESHOLD
C IS=IS-1
IF (IS.LE.0)GO TO 400
K=-(2**IS)
EMAX=EMAX*(2.0**K)
GO TO 300
400 IF (IST.EQ.0)GO TO 500
IF (IS.GE.(-33))GO TO 300
C METHOD HAS FAILED TO CONVERGE
C NERR=-1
C TRANSFER EIGENVALUES AND EXIT
C DO 510 I=1,NN
510 ELAM(I)=A(I,I)
RETURN
END

```



```

C      SUBROUTINE FIXFOR(C, F)
C      IMPLICIT REAL*8(A-H,O-Z)
C      INTEGER C, YES
C      DIMENSION F(1)
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES /, NO/4HNNO /
C
C      GO TO 10, 20, 30, 40, 50, 60, 70, 80), C
C
C      10 CONTINUE
C          F(1) = 0.0D0
C          F(2) = 0.0D0
C          F(7) = 0.0D0
C          F(8) = -F(2)
C          RETURN
C
C      20 CONTINUE
C          F(1) = 0.0D0
C          F(2) = 0.0D0
C          RETURN
C
C      30 CONTINUE
C          F(1) = 0.0D0
C          F(7) = 0.0D0
C          F(8) = -F(2)
C          RETURN
C
C      40 CONTINUE
C          F(1) = 0.0D0
C          F(8) = -F(2)
C          RETURN
C
C      50 CONTINUE
C          F(7) = 0.0D0
C          F(2) = 0.0D0
C          F(8) = 0.0D0
C          RETURN
C
C      60 CONTINUE
C          F(2) = 0.0D0

```



```

F(8) = 0.0DO
RETURN
C   70 CONTINUE
    F(7) = 0.0DO
    F(8) = -F(2)
    RETURN
C   80 CONTINUE
    F(8) = -F(2)
    RETURN
END

```

```

SUBROUTINE ADJUST(I1,I2,QR,QRD,QRDD,Q,QQ,QQD)
C   IMPLICIT REAL*8(A-H,O-Z)
C   INTEGER YES
C   DIMENSION QR(1), QRD(1), QRDD(1), Q(1), QQ(1), QDD(1)
C   COMMON NUNIT, IPRINT
C   DATA YES/4HYES /, NO/4HNNO /
C   QR( I1 ) = Q( I2 )
C   QRD( I1 ) = QD( I2 )
C   QRDD( I1 ) = QDD( I2 )
C
C   RETURN
END

```



```

      SUBROUTINE GPRINT(FPLOT, GRID, LINES, TITLE, YNAME, XSCALE, IP, NPRINT,
1      INTEGER YES
C      DIMENSION YNAME(20), LABEL(1), FPLOT(KTIME) , DEG(1), XSCALE(1),
1      TITLE(1), LABEL(1)
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES /, NO/4HN0 /
C      5000 FORMAT('0',5X,'4(1PE15.7',5X)
C      5001 FORMAT(' ',5X,15X,1P2E15.7)
C      5026 FORMAT('1',18A4)
C      5027 FORMAT('+',72X,15A4)
C      5028 FORMAT('0',40X,18A4)
C      XMIN = DEG(1)
C      XMAX = DEG(KTIME)
C      YMIN = 1.0E40
C      YMAX=-1.E40
C      DO 10 K = 1,KTIME
C      IF(FPLOT(K).GT.YMAX) YMAX = FPLOT(K)
C      IF(FPLOT(K).LT.YMIN) YMIN = FPLOT(K)
10      CONTINUE
      WRITE(NUNIT,5026) (LABEL(JA), JA = 1,18)
      WRITE(NUNIT,5027) (TITLE(JA), JA = 1,18)
      WRITE(NUNIT,5028) (YNAME(JA), JA = 1,18)
C      IRYP1 = 80*IP + 1
1      CALL OLPLT(DEG,FPLOT,KTIME,XMAX,XMIN,YMAX,IRYP1,GRID,XSCALE,
0      ,NPRINT,LINES)
C      IF(NPRINT .EQ. NO) RETURN
C      WRITE(6,5000) XMIN,XMAX,YMIN,YMAX
C      DO 51 JA = 1,KTIME
C      WRITE(6,5001) JA, DEG(JA), FPLOT(JA)
51      CONTINUE
C      RETURN
END

```



```

1      SUBROUTINE GRAPH(FPLOT, YNAME1, YNAME2, XNAME, DEG1, DEG2, Y1, Y2, Y3, Y4,
2          NTIME, LABEL, TITLE, PIN2$8, CLEAR, IPLOT, XSCALE,
3          IP, NPRINT, GRID, LINES)
C
1      INTEGER*2 GRID(LINES, 1)
3      INTEGER YES, PIN2$8
C
1      REAL*8 CLEAR
2      REAL*4 LABEL
C
C      DIMENSION XNAME(1), DEG1(1), DEG2(NTIME,2), XSCALE(1), TITLE(1),
1      FPLOT(NTIME,1), YNAME(20,1), Y1(NTIME,1), Y2(NTIME,1),
2      Y3(NTIME,1), Y4(NTIME,1),
3      YNAME1(20,1), YNAME2(20,1), LABEL(1)
C
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES /, NO/4HNO /
C
1      DO 5 I = 1,10
2      IF(FPLOT(1,I) .EQ. 0.0) FPLOT(1,I) = 1.0E-10
3      CONTINUE
C
C      IF(IDRAW .EQ. NO) GO TO 2501
C
C      4201 CALL DRUB(DEG1,Y1,2,NTIME,10.,6.,1,LABEL,TITLE,XNAME,YNAME1)
1      4202 CALL DRUB(DEG1,Y2,2,NTIME,10.,6.,1,LABEL,TITLE,XNAME,YNAME2)
C      4203 CALL DRUPB(DEG1,FPLOT(1,5),1,NTIME,10.,6.,1,LABEL,TITLE,XNAME,
1      YNAME(1,5))
1      4204 CALL DRUPB(DEG1,FPLOT(1,6),1,NTIME,10.,6.,1,LABEL,TITLE,XNAME,
1      YNAME(1,6))
1      4205 CALL DRUPB(DEG1,FPLOT(1,7),1,NTIME,10.,6.,1,LABEL,TITLE,XNAME,
1      YNAME(1,7))
C
C      IF(PIN2$8 .EQ. YES) GO TO 4207
C      4206 CALL DRUPB(DEG1,FPLOT(1,9),1,NTIME,10.,6.,1,LABEL,TITLE,XNAME,
1      YNAME(1,9))
C
C      4207 IF(CLEAR .LE. -10.0) GO TO 2501
C      4208 CALL DRUPB(DEG1,FPLOT(1,10),1,NTIME,10.,6.,1,LABEL,TITLE,XNAME,
1      YNAME(1,10))

```



```

4209 CALL DRUPB( DEG1, FPLOT(1,11), 1,NTIME,10..,1,LABEL,TITLE,XNAME,
1      YNAME(1,1) ) 00000470
1      CALL DRUG (DEG2,Y4,2,NTIME,10..,1,LABEL,TITLE,XNAME,YNAME(1,13)) 00000480
C      00000500
C      00000510
C      00000520
C      00000530
C      00000540
C      00000550
C      00000560
C      00000570
C      00000580
C      00000590
C      00000600
C      00000610
C      00000620
C      00000630
C      00000640
C      00000650
C      00000660
C      00000670
C      00000680
C      00000690
C      00000700
C      00000710
C      00000720
C      00000730
C      00000740
C      00000750
C      00000760
C      00000770
C      00000780
C      00000790
C      00000800
C      00000810
C      00000820
C      00000830
C      00000840
C      00000850
C      00000860
C      00000870
C      00000880
C      00000890
C      00000900
C      00000910
C      00000920
C      00000930

C 2501 CONTINUE

C
C      IF(IPILOT .EQ. NO) GO TO 4199
C
C      4101 CALL GPRINT(FPLOT(1,1), DEG1, NTIME, YNAME(1, 1), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C
C 4102 CALL GPRINT(FPLOT(1, 2), DEG1, NTIME, YNAME(1, 2), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C 4103 CALL GPRINT(FPLOT(1, 3), DEG1, NTIME, YNAME(1, 3), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C 4104 CALL GPRINT(FPLOT(1, 4), DEG1, NTIME, YNAME(1, 4), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C 4105 CALL GPRINT(FPLOT(1, 5), DEG1, NTIME, YNAME(1, 5), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C 4106 CALL GPRINT(FPLOT(1, 6), DEG1, NTIME, YNAME(1, 6), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C
C 4107 IF(CLEAR LE-10:0) GO TO 4200
C      CALL GPRINT(FPLOT(1, 7), DEG1, NTIME, YNAME(1, 7), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C
C 4200 IF(PIN2$8 EQ YES) GO TO 4109
C      CALL GPRINT(FPLOT(1, 8), DEG1, NTIME, YNAME(1, 8), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C
C 4109 CALL GPRINT(FPLOT(1, 9), DEG1, NTIME, YNAME(1, 9), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C
C 4110 IF(CLEAR LE-10:0) GO TO 4199
C      CALL GPRINT(FPLOT(1,10), DEG1, NTIME, YNAME(1,10), XSCALE, IP,
C      NPRINT, GRID, LINES, TITLE, LABEL)
C
C 4199 CONTINUE
C
C      RETURN
C      END

```



```

SUBROUTINE INITAL(KSYS,Q,QQ,AMPW,FS,QDD,AMPW,FS,FZERO,XZERO,ROCKER,ISYS,
1      COLM,COLC,COLK,DP,F,DAMP,STFRI,NPRINT)
C      IMPLICIT REAL*8(A-H,O-Z)
C      INTEGER YES
C
C      DIMENSION COLM(KSYS),COLC(KSYS),COLK(KSYS),
C      DIMENSION Q(10),QD(1),QDD(1),DP(1),F(1),DAMP(KSYS,KSYS),
C      1      STFRI(ISYS,1),STF(KSYS,KSYS),
C
C      COMMON NUNIT, IPRINT
C
C      DATA YES/4HYES/, NO/4HN0/
C
C      5043 FORMAT('0',10X,' FROM SUBROUTINE INITIAL')
C      5044 FORMAT('0',1P7D18.9,' ')
C      5045 FORMAT('0',1/,20X,1,-,-,STFI,-,-,-,-,1,/)
C      5046 FORMAT('0',1/,20X,1,-,-,STFRI,-,-,-,-,1,/)
C
C      *****
C      *** SET UP INITIAL VALUES
C      *** OF DISPLACEMENT(Q) VELOCITY(QDD),
C      *** AND ACCELERATION(QDD)
C      ***
C      *****
C
C      DO 300 JA = 1, KSYS
C      QD(JA) = -AMPW
C      QDD(JA) = 0.0D0
C      300 CONTINUE
C
C      QD(1) = +AMPW
C      QD(2) = +AMPW
C      QD(4) = -AMPW*ROCKER
C      QD(7) = +AMPW*ROCKER
C      QD(8) = +AMPW
C
C      Q(1) = XZERO
C
C      DO 350 JA = 1, ISYS
C      DP(JA) = -(COLM(JA)*QDD(1)+COLC(JA)*QD(1)+COLK(JA)*Q(1))
C
C      DO 350 JA = 1, ISYS
C      DP(JA) = DP(JA)-DAMP(JA,JB)*QD(JB+1)
C      STFRI(JA,JB) = STF(JA,JB)

```



```

C 350 CONTINUE
C      DO 8 I = 0, KSYS
C      F(I) = 0.0D0
C      CONTINUE
C
C*****CALL DINVS(STFRI,ISYS,ISYS)
C
C      ** CALL SUBROUTINE DINVS
C      ** DINVS CALCULATES THE INVERSE STIFFNESS
C      ** MATRIX
C      ** SEND STF - RETURN STFRI
C
C*****CALL DINVS(STFRI,ISYS,ISYS)
C
C      DO 360 JA = 1, ISYS
C      Q(JA+1) = 0.
C
C      DO 360 JB = 1, ISYS
C      Q(JB+1) = Q(JA+1)+STFRI(JA,JB)*DP(JB)
C
C      360 CONTINUE
C
C      CALL ADJUST(1,2, Q, QD, QDD, Q, QD, QDD)
C      CALL ADJUST(8,2, Q, QD, QDD, Q, QD, QDD)
C
C      RETURN
C

```



```

SUBROUTINE INPUT(LABEL, TITLE, XNAME, YNAME, CARDS, TAPE, IPLOT, IDRAW,
1      AGTOUT, RPM, CLEAR, WTDEG, AMP, DELT,
2      KPRINT, DAMPNNG, AXILTH, REALTH,
3      ROCKER, AXILTH, REALTH,
4      NTIME, KSYS, NCLEAR, NC,
5      NCOUNT, IDEG, PIN2$8, MPRINT)
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C
C      REAL*4 LABEL, TITLE, XNAME, YNAME,
1      VLV, TWO1, TWO2
C
C      INTEGER YES,AGTOUT,DAMPNNG,TAPE,CARDS,SEAT,PIN2$8
C
C      DIMENSION YNAME(20,1),AXILTH(1),REALTH(1),
1      LABEL(1),TITLE(1),XNAME(1)
C
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES/, NO/4HNNO/
C      DATA TWO1//, TWO2//, NO C/, VLV/'NO V'/
C
C      5001 FORMAT(18A4)
5002 FORMAT(1A4)
5003 FORMAT(D10.3)
5004 FORMAT(I4)
5005 FORMAT(1A4,2I4)
5006 FORMAT(4D10.3)
5007 FORMAT('0',10(I4,8X))
5008 FORMAT('0',6(D10.3,5X))
5009 FORMAT('0',8(A4,10X))
5010 FORMAT(D10.3,2X,1A4)
C
C      REALTH(1) = 9.0DO
      REALTH(2) = 1.0DO
      REALTH(3) = 1.5DO
      REALTH(4) = 36.0DO
      AXILTH(1) = 1.0DO
      AXILTH(2) = 3.4DO
      AXILTH(3) = 5.1DO
      AXILTH(4) = 1.0DO
      AMP = 0.5DO
      DAMPNNG = YES
      KSYS = 8
      NC = 8
      NCLEAR = 3
      NUNIT = 6

```


C C C C C C C C C

000000950
000000960
000000970
000000980
000000990
000001000
000001010
000001020
000001030
000001040
000001050
000001060
000001070
000001080
000001090
000001100

C C RETURN
C 50 IF(IUNIT .EQ. 4) END FILE 4
IF(IUNIT .EQ. 4) END FILE 4
IF(IUNIT .EQ. 4) REWIND 4
C STOP
C CCCCCCCCC
END


```

1      SUBROUTINE MATRIX( COLM, COLC, COLK, SYSM, DAMP, STF, IDIM, KSYS,
1      KSYS1, C, IQ, MPRINT, LAYER, IDIM, KSYS, 00000010
1      00000020
1      00000030
1      00000040
1      00000050
1      00000060
1      00000070
1      00000080
1      00000090
1      00000100
1      00000110
1      00000120
1      00000130
1      00000140
1      00000150
1      00000160
1      00000170
1      00000180
1      00000190
1      00000200
1      00000210
1      00000220
1      00000230
1      00000240
1      00000250
1      00000260
1      00000270
1      00000280
1      00000290
1      00000300
1      00000310
1      00000320
1      00000330
1      00000340
1      00000350
1      00000360
1      00000370
1      00000380
1      00000390
1      00000400
1      00000410
1      00000420
1      00000430
1      00000440
1      00000450
1      00000460

C      IMPLICIT REAL*8(A-H,O-Z)

C      INTEGER YES, C

C      DIMENSION COLM(KSYS,1), COLC(KSYS,1), COLK(KSYS,1),
1      DAMP(KSYS,KSYS,1), STF(KSYS,KSYS,1), STFI(KSYS1,1),
2      SYSM(KSYS,KSYS,1), IQ(KSYS,1), LAYER(1), IDIM(1)

C      COMMON NUNIT, IPRINT

C      DATA YES/4HYES/, NO/4HNO/ /

KCOUNT = 1
LC = LAYER(C)
***** ESTABLISH THE 1X6 COLUMN MATRIX FROM THE
***** SYSTEM M-C-K MATRICES WITHOUT ROW 1 *****
***** ESTABLISH THE 6X6 MATRICES FROM THE
***** SYSTEM M-C-K MATRICES WITHOUT ROW/COL 7 *****
***** COUNT = 1
***** COUNT = 2
***** COUNT = 3
***** COUNT = 4
***** COUNT = 5
***** COUNT = 6
***** COUNT = 7
***** COUNT = 8
***** COUNT = 9
***** COUNT = 10
***** COUNT = 11
***** COUNT = 12
***** COUNT = 13
***** COUNT = 14
***** COUNT = 15
***** COUNT = 16
***** COUNT = 17
***** COUNT = 18
***** COUNT = 19
***** COUNT = 20
***** COUNT = 21
***** COUNT = 22
***** COUNT = 23
***** COUNT = 24
***** COUNT = 25
***** COUNT = 26
***** COUNT = 27
***** COUNT = 28
***** COUNT = 29
***** COUNT = 30
***** COUNT = 31
***** COUNT = 32
***** COUNT = 33
***** COUNT = 34
***** COUNT = 35
***** COUNT = 36
***** COUNT = 37
***** COUNT = 38
***** COUNT = 39
***** COUNT = 40
***** COUNT = 41
***** COUNT = 42
***** COUNT = 43
***** COUNT = 44
***** COUNT = 45
***** COUNT = 46

20 DO 200 JA=1,KSYS
1F(JC,EQ.0) GO TO 300
C      DO 100 JB=1,KSYS
1F(JD,EQ.0) GO TO 150
C      SYSM(JA,JB,C) = SYSM(JC,JD,LC)
DAMP(JA,JB,C) = DAMP(JC,JD,LC)
STF(JA,JB,C) = STF(JC,JD,LC)
100 CONTINUE
150 IDIM(C) = KCOUNT

```



```

KCOUNT = KCOUNT + 1
C   JCIQ = JC
C   IFOR = 1
C   IF(C .LE. 4) IFOR = 7
C   IF(C .EQ. 1) JCIQ = IQ(JA,6)
C   IF(C .EQ. 3) JCIQ = IQ(JA,8)
C
C   COLM(JA,C) = SYSM(JCIQ,IFOR,LC)
C   COLC(JA,C) = DAMP(JCIQ,IFOR,LC)
C   COLK(JA,C) = STF(JCIQ,IFOR,LC)
C
C   200 CONTINUE
C   IF(MPRINT .EQ. NO) GO TO 350
C   CALL PRNTMX( COLM(1,C),COLC(1,C),COLK(1,C),SYSM(1,C),
C   1           DAMP(1,1,C),STF(1,1,C),KSYS, C, IDIM(C) )
C
C   350 IF(C .GT. 1) RETURN
C   DO 400 JA = 1,KSYSM1
C   DO 400 JB = 1,KSYSM1
C   STFI(JA,JB) = STF(JA,JB,3)
C
C   400 CONTINUE
C
C   *****
C   *** CALL SUBROUTINE DINVS
C   *** DINVS CALCULATES THE INVERSE STIFFNESS
C   *** MATRIX
C   *** SEND STF -- RETURN STFI
C   ***
C   ****
C   CALL DINVS(STFI,KSYSM1,KSYSM1)
C   RETURN
C   END

```


SUBROUTINE CODED BY DR. R. C. WINFREY AND R. ANDERSON
MODIFIED BY C. W. GNIKA

```

1 SUBROUTINE MCK( SYSM, DAMP, STF, KSYS, JC, N, NPRINT, COFD, PHI,
2 ELAM, C)
3 IMPLICIT REAL*8(A-H,O-Z)
4
5 INTEGER YES, C
6
7 DIMENSION PHTINV(8,8), PHITM(8,8), PHITK(8,8), DIAGM(8,8),
8 E(5) SEE(8,8), PHINV(8,8), PHI(KSYS,KSYS(1),DIAGM(8,8),
9 SYSM(KSYS,KSYS,1),PHI(KSYS,KSYS(1)),DAMP(KSYS,KSYS(1)),
10 DIAGK(8,8),DIAGC(8,8),ZTA(8,8),BETA(8,8),L(8),Y(5),
11 AM(5),AMM(5),DIA(5),ELAM(KSYS,1),PHTIDC(8,8),
12 COMMON NUNIT, IPRINT
13 DATA YES/4HYES/, NO/4HN0/
14 LENGTH Y/9.000, 1.000, 1.500, 36.000, 3.000/
15 DATA AMM/5*0.283D0/, E/5*3.0D7/
16 DATA DIA/0.3D0, 0.6D0, 0.6D0, 0.07D0, 0.3D0/
17 DATA PI/3.14159265359D0/, GRAV/386.0D0/
18
19 FORMAT('0','//',10X,'ERROR RETURN FROM DVVS. NERR = ', I3,/// )
20 FORMAT('4X, 8(1PD15.8, 1X)', C
21 IF(NPRINT .EQ. YES) WRITE(NUNIT,700) (ZTA(KA), KA = 1,JC)
22 RAD=PI/180.0D0
23 N IS THE NUMBER OF ELEMENTS
24 JC = SIZE OF K (OR M) AND NUMBER OF COL OF B
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338

```



```

C      DO 5 I = 1, N
C      AM(I) = AMM(I) * PI * DIA(I)**2 * Y(I) / (4.0D0 * GRAV)
C      5 CONTINUE
C      DO 10 I = 1, JC
C      ZTA(I) = 0.1D0
C
C      IF(C .EQ. 1) ZTA(4) = 0.5D0
C      IF(C .EQ. 3) ZTA(1) = 0.5D0
C      IF(C .EQ. 3) ZTA(4) = 0.1D0
C      SET SYSTEM K (OR M) TO 0
C
C      DO 10 J = 1, JC
C      SYSM(I,J,C) = 0.0D0
C      STF(I,J,C) = 0.0D0
C      10 CONTINUE
C
C      CALCULATE EACH ELEMENT K (OR M)
C      APYX = PI * E(3) * DIA(3)**4 / 64.0D0
C      APYY = PI * E(2) * DIA(2)**4 / 64.0D0
C
C      ONE   = E(1) * PI * DIA(1)**2 / (Y(1)**4.0D0 )
C      TWO   = 12.0D0 * APYY / (Y(2)**3 )
C      THREE = 6.0D0 * APYY / (Y(2)**2 )
C      FOUR  = 4.0D0 * APYY / (Y(2) )
C      FIVE  = 2.0D0 * APYY / (Y(2) )
C      SIX   = 12.0D0 * APYY / (Y(3)**3 )
C      SEVEN = E(5) * PI * DIA(5)**2 / (Y(5)**4.0D0 )
C      EIGHT = 6.0D0 * APYY / (Y(3)**2 )
C      NINE  = 4.0D0 * APYY / (Y(3) )
C      TEN   = E(4) * PI * DIA(4)**2 / (Y(4)**4.0D0 )
C      ELVEN = 2.0D0 * APYY / (Y(3) )
C
C      STF(1,1,C) = ONE
C      STF(1,2,C) = -ONE
C      STF(2,2,C) = TWO
C      STF(2,3,C) = THREE
C      STF(2,6,C) = FOUR
C      STF(3,3,C) = FIVE
C      STF(4,4,C) = SIX + SEVEN + TEN
C      STF(4,5,C) = -EIGHT
C      STF(4,6,C) = -NINE
C      STF(4,7,C) = SEVEN
C      STF(5,5,C) = ANINE
C      STF(5,6,C) = ELVEN

```



```

STF(6,6,C) = ANINE + FOUR
STF(7,7,C) = SEVEN
STF(8,8,C) = ONE

C   SYSM(1,1,C) = AM(1) / 3.00D0
    SYSM(1,8,C) = AM(1) / 6.00D0
    SYSM(2,2,C) = 156.00D0 * AM(2) / 420.00D0
    SYSM(2,3,C) = AM(2) * 22.00D0 * Y(2) / 420.00D0
    SYSM(2,6,C) = -AM(2) * 13.00D0 * Y(2) ** 2 / 420.00D0
    SYSM(3,3,C) = AM(2) * 4.00D0 * Y(2) ** 2 / 420.00D0
    SYSM(3,6,C) = -AM(2) * 3.00D0 * Y(2) ** 2 / 420.00D0
    SYSM(3,7,C) = AM(3) * 156.00D0 * ( AM(5) + AM(4) ) / 3.0D0
    SYSM(4,4,C) = AM(3) * 22.00D0 * Y(3) / 420.00D0
    SYSM(4,5,C) = -AM(3) * 22.00D0 * Y(3) / 420.00D0
    SYSM(4,6,C) = AM(3) * 13.00D0 * Y(3) / 420.00D0
    SYSM(4,7,C) = -AM(5) / 6.00D0 * Y(3) ** 2 / 420.00D0
    SYSM(5,5,C) = AM(3) * 4.00D0 * Y(3) ** 2 / 420.00D0
    SYSM(5,6,C) = -AM(3) * 3.00D0 * Y(3) ** 2 / 420.00D0
    SYSM(5,7,C) = SYSM(3,3,C) + SYSM(5,5,C)
    SYSM(8,8,C) = AM(1) / 3.00D0

C   IF(C .EQ. 1) GO TO 12
    STF(1,2,3) = -ONE + TWO
    SYSM(1,2,3) = AM(1) / 6.00D0 + 156.00D0*AM(2)/420.0D0
    SYSM(2,2,3) = AM(1) / 3.00D0 + 156.00D0*AM(2)/420.0D0

C   ZTA(1) = 0.1D0
    ZTA(4) = 0.5D0
12  CONTINUE
    DO 20 JA = 1, JC

C   15  DO 20 JB = 1, JC
        STF(JB,JA,C) = STF(JA,JB,C)
        SYSM(JB,JA,C) = SYSM(JA,JB,C)
        DIAGC(JA,JB) = SYSM(JA,JB,C)
        PHINV(JA,JB) = STF(JA,JB,C)
20  CONTINUE

C   24 CALL DVVS( PHIINV, DIAGC, ELAM(1,C), PHI(1,1,C), JC, JC, JC,
1      KSYS, NERR)
C   IF( NERR .NE. 0) WRITE(NUNIT,330) NERR
C   25 DO 37 JA = 1, JC
    ELAM(JA,C) = ELAM(JA,C)**.5
37  CONTINUE

```



```

C      DO 1310   JA = 1, JC
C      35  DO 1310   JB = 1, JC
      PHTINV(JA,JB) = PHI(JB,JA,C)
      PHINV(JA,JB) = PHI(JA,JB,C)
C      DIAGC(JA,JB) = 0.0D0
C      1310  CONTINUE
C      1312  DO 1315   KA = 1, JC
C      1314  DO 1315   KB = 1, JC
      PHITK(KA,KB) = 0.0D0
      PHITM(KA,KB) = 0.0D0
C      DO 1315   KC = 1, JC
      PHITM(KA,KB) = PHTINV(KA,KC)*SYSM(KC,KB,C) + PHITM(KA,KB)
      PHITK(KA,KB) = PHTINV(KA,KC)*STF(KC,KB,C) + PHITK(KA,KB)
C      1315  CONTINUE
C      DO 1318   KA = 1, JC
C      DO 1318   KB = 1, JC
      DIAGK(KA,KB) = 0.0D0
      DIAGM(KA,KB) = 0.0D0
C      1316  DO 1318   KC = 1, JC
      DIAGM(KA,KB) = PHITM(KA,KC)*PHI(KC,KB,C) + DIAGM(KA,KB)
      DIAGK(KA,KB) = PHITK(KA,KC)*PHI(KC,KB,C) + DIAGK(KA,KB)
C      1318  CONTINUE
C      DO 1328   KA = 1, JC
      DIAGC(KA,KA) = 2.0D0*ZTA(KA)*( DIAGM(KA,KA)*DIAGK(KA,KA) )**0.5D0
C      1328  CONTINUE
C      1326  CALL DINVS(PHTINV,JC,KSYS)

```



```

C 1327 CALL DINVS(PHINV,JC,KSYS)
C
C DO 1322 KA = 1,JC
C DO 1322 KB = 1,JC
PHTIDC(KA,KB) = 0.0DO
C 1323 DO 1322 KC = 1,JC
PHTIDC(KA,KB) = PHTINV(KA,KC)*DIAGC(KC,KB) + PHTIDC(KA,KB)
C 1322 CONTINUE
C DO 1324 KA = 1,JC
C DO 1324 KB = 1,JC
SEE(KA,KB) = 0.0DO
C 1321 DO 1324 KC = 1,JC
SEE(KA,KB) = PHTIDC(KA,KC)*PHINV(KC,KB) + SEE(KA,KB)
C 1324 CONTINUE
C DO 1325 KA = 1,JC
DO 1325 KB = 1,JC
DAMP(KA,KB,C) = SEE(KA,KB)*COFD
C 1325 CONTINUE
C RETURN
END

```



```

      SUBROUTINE MONITOR(NCLEAR, CONFIG, ICORD, F1, C, NPRINT, IDT, DTP,
1      DTPMIN, IMPACT, Q, QSAV, FSAV, DT)
C      IMPLICIT REAL*8(A-H,O-Z)
C      INTEGER YES, C, CONFIG(16,1)
C      DIMENSION ICORD(NCLEAR,1), F(1), FSAV(1), DTP(1), Q(1), QSAV(1),
1      IMPACT(1)
C      COMMON NUNIT, IPRINT
C      DATA YES/4HYES /, NO/4HNO /
C      5000 FORMAT('0','20X','MONITOR INDICATES MORE THAN ONE CONFIGURATION
1      CHANGE IN ONE TIME STEP')
5001 FORMAT('0','20X','***** ERROR -- COMPUTER TRUNCATION -- KK
1      NOT CORRECTLY SET *****')
5002 FORMAT('0','10X','DTP(1)=',I2,',',I2,',',I2,',',I2,',',I2,',',I2,',',
1      INDEX OF DTP(1) FOR MIN IS ',I2,', AND IMPACT = ',I2,',',I2,',',I2,',',
1      I2)
C      IDT = 0
EPS1 = 1.0D-1
EPS2 = 1.0D-6
C      DO 20 I = 1,0D0
DTP(I) = +1.0D0
IMPACT(I) = 0
CONTINUE
20 CONTINUE
C      DO 23 KK = 1, NCLEAR
INDEX = ICORD(KK,1)
NDEX = ICORD(KK,2)
C      IF( CONFIG(C,KK) .EQ. 0 ) GO TO 12
C      ****
**      FORCE MONITOR
**      ****
C      IF(F(INDEX) .GE. 0.0) GO TO 23

```



```

      FSAVE = FSAV( INDEX )
      FSEND = F( INDEX )

      CALL DTPRIM( FSEND, KK, FSAVE, DT, DTP, NPRINT )
      IDT = IDT + 1
      GO TO 23

      *****
      ** DISPLACEMENT MONITOR
      *****

12   DISPL = Q( INDEX ) - Q( INDEX )
      IF(DISPL .GE. 0.0) GO TO 23
      IF( DABS(DISPL) .LE. EPS2) GO TO 23
      GO TO 23

      IMPACT(KK) = KK
      DOLD = QSAV( INDEX ) - QSAV( INDEX )
      IF(DOLD .LE. 0.0) DOLD = 1.0D-10

      CALL DTPRIM( DISPL, KK, DOLD, DT, DTP, NPRINT )
      IDT = IDT + 1

23   CONTINUE

      DO 25 I = 1, NCLEAR
      IF(DTP(I) .LT. 0.1DD0) GO TO 50
      CONTINUE
      DT = -1

      RETURN

25   CONTINUE
      DT = -1

      IF(IDT .GT. 1 .AND. NPRINT .EQ. YES) WRITE(
      DO 60 KK = 1, NCLEAR
      DTPMIN = DMIN( DTPMIN, DTP(I) )
      IF(NPRINT .EQ. YES) WRITE(NUNIT,5002) JA, DTP(J)
      CONTINUE

      IF(IDT .GT. 1 .AND. NPRINT .EQ. YES) WRITE(
      DO 70 KK = 1, NCLEAR
      IF(DTPMIN .EQ. DTP(KK)) GO TO 80
      CONTINUE

      WRITE(NUNIT,5001)

```



```

00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030
00001040
00001050
00001060
00001070
00001080
00001090

***** CHANGE CONFIGURATION ****
***   IF(NPRINT.EQ.NO) RETURN
***   WRITE(NUNIT,5003) KK, IMPACT(KK)
***   RETURN
END

```

```

00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
00000180
00000190

SUBROUTINE CAM(WT,AMP,AMPW,XZERO,W2,Q,QQD,QDD)
C IMPLICIT REAL*8(A-H,O-Z)
C INTEGER YES
C DIMENSION Q(1), QD(1), QDD(1)
C COMMON NUNIT, IPRINT
C DATA YES/4HYES/, NO/4HNO/
C AMP$IN = AMP*D$IN(WT)
C Q(9) = AMP$IN + XZERO
C QD(9) = AMPW*D$COS(WT)
C QDD(9) = AMP$IN*W2
C RETURN
END

```



```

SUBROUTINE NEWMRK( ISYS, KSYS, QDD, BMIV, DA,
1 DT, DT02, DT203, Q, QD, QDD, QD15, DV, NC,
2 ALFA, DT02, DT206, CONFIG, F, ICORD, NCLEAR, NPRINT, IQ, COLM, COLC, COLK, IDP,
3 PIN2$8, IDIM, LAYER, IGU, QDDSAV, QDSAV, SCAL, IMP)
4
5 IMPLICIT REAL*8(A-H,O-Z)
6
7 C
8 C
9 C
10 C
11 C
12 C
13 C
14 C
15 C
16 C
17 C
18 C
19 C
20 C
21 C
22 C
23 C
24 C
25 C
26 C
27 C
28 C
29 C
30 C
31 C
32 C
33 C
34 C
35 C
36 C
37 C
38 C
39 C
40 C
41 C
42 C
43 C
44 C
45 C
46 C
47 C
48 C
49 C
50 C
51 C
52 C
53 C
54 C
55 C
56 C
57 C
58 C
59 C
60 C
61 C
62 C
63 C
64 C
65 C
66 C
67 C
68 C
69 C
70 C
71 C
72 C
73 C
74 C
75 C
76 C
77 C
78 C
79 C
80 C
81 C
82 C
83 C
84 C
85 C
86 C
87 C
88 C
89 C
90 C
91 C
92 C
93 C
94 C
95 C
96 C
97 C
98 C
99 C
100 C
101 C
102 C
103 C
104 C
105 C
106 C
107 C
108 C
109 C
110 C
111 C
112 C
113 C
114 C
115 C
116 C
117 C
118 C
119 C
120 C
121 C
122 C
123 C
124 C
125 C
126 C
127 C
128 C
129 C
130 C
131 C
132 C
133 C
134 C
135 C
136 C
137 C
138 C
139 C
140 C
141 C
142 C
143 C
144 C
145 C
146 C
147 C
148 C
149 C
150 C
151 C
152 C
153 C
154 C
155 C
156 C
157 C
158 C
159 C
160 C
161 C
162 C
163 C
164 C
165 C
166 C
167 C
168 C
169 C
170 C
171 C
172 C
173 C
174 C
175 C
176 C
177 C
178 C
179 C
180 C
181 C
182 C
183 C
184 C
185 C
186 C
187 C
188 C
189 C
190 C
191 C
192 C
193 C
194 C
195 C
196 C
197 C
198 C
199 C
200 C
201 C
202 C
203 C
204 C
205 C
206 C
207 C
208 C
209 C
210 C
211 C
212 C
213 C
214 C
215 C
216 C
217 C
218 C
219 C
220 C
221 C
222 C
223 C
224 C
225 C
226 C
227 C
228 C
229 C
230 C
231 C
232 C
233 C
234 C
235 C
236 C
237 C
238 C
239 C
240 C
241 C
242 C
243 C
244 C
245 C
246 C
247 C
248 C
249 C
250 C
251 C
252 C
253 C
254 C
255 C
256 C
257 C
258 C
259 C
260 C
261 C
262 C
263 C
264 C
265 C
266 C
267 C
268 C
269 C
270 C
271 C
272 C
273 C
274 C
275 C
276 C
277 C
278 C
279 C
280 C
281 C
282 C
283 C
284 C
285 C
286 C
287 C
288 C
289 C
290 C
291 C
292 C
293 C
294 C
295 C
296 C
297 C
298 C
299 C
300 C
301 C
302 C
303 C
304 C
305 C
306 C
307 C
308 C
309 C
310 C
311 C
312 C
313 C
314 C
315 C
316 C
317 C
318 C
319 C
320 C
321 C
322 C
323 C
324 C
325 C
326 C
327 C
328 C
329 C
330 C
331 C
332 C
333 C
334 C
335 C
336 C
337 C
338 C
339 C
340 C
341 C
342 C
343 C
344 C
345 C
346 C
347 C
348 C
349 C
350 C
351 C
352 C
353 C
354 C
355 C
356 C
357 C
358 C
359 C
360 C
361 C
362 C
363 C
364 C
365 C
366 C
367 C
368 C
369 C
370 C
371 C
372 C
373 C
374 C
375 C
376 C
377 C
378 C
379 C
380 C
381 C
382 C
383 C
384 C
385 C
386 C
387 C
388 C
389 C
390 C
391 C
392 C
393 C
394 C
395 C
396 C
397 C
398 C
399 C
400 C
401 C
402 C
403 C
404 C
405 C
406 C
407 C
408 C
409 C
410 C
411 C
412 C
413 C
414 C
415 C
416 C
417 C
418 C
419 C
420 C
421 C
422 C
423 C
424 C
425 C
426 C
427 C
428 C
429 C
430 C
431 C
432 C
433 C
434 C
435 C
436 C
437 C
438 C
439 C
440 C
441 C
442 C
443 C
444 C
445 C
446 C
447 C
448 C
449 C
450 C
451 C
452 C
453 C
454 C
455 C
456 C
457 C
458 C
459 C
460 C
461 C
462 C
463 C
464 C
465 C
466 C
467 C
468 C
469 C
470 C
471 C
472 C
473 C
474 C
475 C
476 C
477 C
478 C
479 C
480 C
481 C
482 C
483 C
484 C
485 C
486 C
487 C
488 C
489 C
490 C
491 C
492 C
493 C
494 C
495 C
496 C
497 C
498 C
499 C
500 C
501 C
502 C
503 C
504 C
505 C
506 C
507 C
508 C
509 C
510 C
511 C
512 C
513 C
514 C
515 C
516 C
517 C
518 C
519 C
520 C
521 C
522 C
523 C
524 C
525 C
526 C
527 C
528 C
529 C
530 C
531 C
532 C
533 C
534 C
535 C
536 C
537 C
538 C
539 C
540 C
541 C
542 C
543 C
544 C
545 C
546 C
547 C
548 C
549 C
550 C
551 C
552 C
553 C
554 C
555 C
556 C
557 C
558 C
559 C
559 C
560 C
561 C
562 C
563 C
564 C
565 C
566 C
567 C
568 C
569 C
570 C
571 C
572 C
573 C
574 C
575 C
576 C
577 C
578 C
579 C
580 C
581 C
582 C
583 C
584 C
585 C
586 C
587 C
588 C
589 C
589 C
590 C
591 C
592 C
593 C
594 C
595 C
596 C
597 C
598 C
599 C
599 C
600 C
601 C
602 C
603 C
604 C
605 C
606 C
607 C
608 C
609 C
609 C
610 C
611 C
612 C
613 C
614 C
615 C
616 C
617 C
618 C
619 C
619 C
620 C
621 C
622 C
623 C
624 C
625 C
626 C
627 C
628 C
629 C
629 C
630 C
631 C
632 C
633 C
634 C
635 C
636 C
637 C
638 C
639 C
639 C
640 C
641 C
642 C
643 C
644 C
645 C
646 C
647 C
648 C
649 C
649 C
650 C
651 C
652 C
653 C
654 C
655 C
656 C
657 C
658 C
659 C
659 C
660 C
661 C
662 C
663 C
664 C
665 C
666 C
667 C
668 C
669 C
669 C
670 C
671 C
672 C
673 C
674 C
675 C
676 C
677 C
678 C
679 C
679 C
680 C
681 C
682 C
683 C
684 C
685 C
686 C
687 C
688 C
689 C
689 C
690 C
691 C
692 C
693 C
694 C
695 C
696 C
697 C
698 C
699 C
699 C
700 C
701 C
702 C
703 C
704 C
705 C
706 C
707 C
708 C
709 C
709 C
710 C
711 C
712 C
713 C
714 C
715 C
716 C
717 C
718 C
719 C
719 C
720 C
721 C
722 C
723 C
724 C
725 C
726 C
727 C
728 C
729 C
729 C
730 C
731 C
732 C
733 C
734 C
735 C
736 C
737 C
738 C
739 C
739 C
740 C
741 C
742 C
743 C
744 C
745 C
746 C
747 C
748 C
749 C
749 C
750 C
751 C
752 C
753 C
754 C
755 C
756 C
757 C
758 C
759 C
759 C
760 C
761 C
762 C
763 C
764 C
765 C
766 C
767 C
768 C
769 C
769 C
770 C
771 C
772 C
773 C
774 C
775 C
776 C
777 C
778 C
779 C
779 C
780 C
781 C
782 C
783 C
784 C
785 C
786 C
787 C
788 C
789 C
789 C
790 C
791 C
792 C
793 C
794 C
795 C
796 C
797 C
798 C
799 C
799 C
800 C
801 C
802 C
803 C
804 C
805 C
806 C
807 C
808 C
809 C
809 C
810 C
811 C
812 C
813 C
814 C
815 C
816 C
817 C
818 C
819 C
819 C
820 C
821 C
822 C
823 C
824 C
825 C
826 C
827 C
828 C
829 C
829 C
830 C
831 C
832 C
833 C
834 C
835 C
836 C
837 C
838 C
839 C
839 C
840 C
841 C
842 C
843 C
844 C
845 C
846 C
847 C
848 C
849 C
849 C
850 C
851 C
852 C
853 C
854 C
855 C
856 C
857 C
858 C
859 C
859 C
860 C
861 C
862 C
863 C
864 C
865 C
866 C
867 C
868 C
869 C
869 C
870 C
871 C
872 C
873 C
874 C
875 C
876 C
877 C
878 C
879 C
879 C
880 C
881 C
882 C
883 C
884 C
885 C
886 C
887 C
888 C
889 C
889 C
890 C
891 C
892 C
893 C
894 C
895 C
896 C
897 C
898 C
899 C
899 C
900 C
901 C
902 C
903 C
904 C
905 C
906 C
907 C
908 C
909 C
909 C
910 C
911 C
912 C
913 C
914 C
915 C
916 C
917 C
918 C
919 C
919 C
920 C
921 C
922 C
923 C
924 C
925 C
926 C
927 C
928 C
929 C
929 C
930 C
931 C
932 C
933 C
934 C
935 C
936 C
937 C
938 C
939 C
939 C
940 C
941 C
942 C
943 C
944 C
945 C
946 C
947 C
948 C
949 C
949 C
950 C
951 C
952 C
953 C
954 C
955 C
956 C
957 C
958 C
959 C
959 C
960 C
961 C
962 C
963 C
964 C
965 C
966 C
967 C
968 C
969 C
969 C
970 C
971 C
972 C
973 C
974 C
975 C
976 C
977 C
978 C
979 C
979 C
980 C
981 C
982 C
983 C
984 C
985 C
986 C
987 C
988 C
989 C
989 C
990 C
991 C
992 C
993 C
994 C
995 C
996 C
997 C
998 C
999 C

```



```

DP(I) = 0.0D0
50 CONTINUE
C IF(C .EQ. 1 .OR. C .EQ. 3) GO TO 500
C DO 300 JA = 1, NCLEAR
C KDP = IDP(C,JA)
C IF(KDP .NE. 0) GO TO 325
C 300 CONTINUE
C 325 KC = C
C 350 ACCEL = QDD(KDP)
C VEL = QD(KDP)
C DIST = Q(KDP)
C
C DO 400 JA = 1, ISYS
C CSYSM = COLM(JA,KC)
C CDAMP = COLC(JA,KC)
C CSTFF = COLK(JA,KC)
C DP(JA) = CSYSM*ACCEL + CDAMP*VEL + CSTFF*DIST + DP(JA)
C 400 CONTINUE
C 425 IF(KC .NE. C) GO TO 500
C IF(C .EQ. 6) KC = 1
C IF(C .EQ. 8) KC = 3
C KDP = 7
C IF(KC .NE. C) GO TO 350
C 500 CONTINUE
C 600 DO 700 JA = 1, ISYS
C IA = IQ(JA,C)
C ACCEL = QDDSAV(IA)
C VEL = QDSAV(IA)
C
C ALFA(JA) = ACCEL*D102 + VEL*DT + QSAV(IA)
C BETA(JA) = ACCEL*D1203 + VEL*DT + QSAV(IA)
C 700 CONTINUE
C IF(NPRINT .EQ. NO) GO TO 750
C
C WRITE(NUNIT,5008) ( ALFA(JA), BETA(JA), JA = 1, ISYS )
C WRITE(NUNIT,5009) ( BETA(JA), JA = 1, ISYS )
C WRITE(NUNIT,5011)
C
C 750 DO 900 JA = 1, ISYS

```



```

IA = IQ(JA,C)
QDD(IA) = 0.000
C      DO 800   JB = 1, ISYS
      QDD(IA) = QDD(JA) - BMIV(JA,JB)*DP(JB) - DA(JA,JB)*ALFA(JB)
      1 - DV(JA,JB)*BETA(JB)
C      IF(NPRINT.EQ.NO) GO TO 800
      WRITE(UNIT,5010) BMIV(JA,JB), DP(JB), DA(JA,JB), ALFA(JB), DV(JA,JB),
      1 BETA(JB), QDD(JA)
C      800 CONTINUE
C      IF(NPRINT.EQ. YES) WRITE(UNIT,5015)
900 CONTINUE
C      IF(IMP .NE. 2) GO TO 950
      QDD(2) = (SCALE - 1.000) * QDSAV(2) / DT
      WRITE(UNIT,5045) QDD(2), SCALE, QDSAV(2)
      IMP = -10
      DO 1000 JA = 1, ISYS
      IA = IQ(JA,C)
      QD(IA) = ALFA(JA) + QDD(IA)*DT02
      Q(IA) = BETA(JA) + QDD(IA)*DT206
C      1000 CONTINUE
      IF(IMP .NE. -10) GO TO 1050
      WRITE(UNIT,5046) QD(2)
      5046 FORMAT(10X,'NEW CALCULATED VELOCITY OF COORD. TWO IS ',1PD15.7)
      1 IMP = C
      1050 CONTINUE
C      LC = LAYER(C)
      IF(LC.NE.Q.*3) GO TO 1170
      1150 IF(PIN2$8.EQ. NO) GO TO 1175
C      1170 CALL ADJUST(8,2,Q,QD,QDD,Q,LC)
      LC = 3
      1175 CONTINUE
      IF(IGU.EQ.YES) LC = 1
      ILC = IDIM(LC)
C      DO 1300 IA = 1, NCLEAR
      KK = ICORD(IA,1)
      F(KK) = 0.0DD0
      1300 1370
      1380
      1390
      1400
      1410
      1420
C      DO 1200 JA = 1, ILC

```


00001430
00001440
00001450
00001460
00001470
00001480
00001490
00001500
00001510
00001520
00001530
00001540
00001550
00001560
00001570
00001580
00001590
00001600

F(KK) = F(KK) + SYSM(KK,JA,LC)*QDD(JA)
1 + DAMP(KK,JA,LC)*QD(JA) + STF(KK,JA,LC)*Q(JA)

C 1200 CONTINUE

C 1300 CONTINUE
C 1350 IF(PIN2\$8 .EQ. NO) GO TO 1375

C F(2) = 0.000
C F(8) = 0.000

C 1375 CONTINUE
C IF(NPRINT EQ NO) RETURN
C WRITE(NUNIT,5037) F(JQ), JQ = 1, KSYS
C WRITE(NUNIT,5042)
C RETURN
C END

30 CONTINUE

C PLOTTING Y AND X AXIS , IF NECESSARY

```

YRANGE=YMAX-YMIN
XRANGE=XMAX-XMIN
IF (X RANGE.NE.0.) GO TO 298
XMIN=YMIN
XMAX=YMAX
XRANGE=YRANGE
GO TO 299
IF (Y RANGE.NE.0..) GO TO 299
YMIN=XMIN
YMAX=XMAX
XRANGE=XRANGE

```

C BLANKING OUT MATRIX-(GRID)

```

299 DO 300 I=1,IRYP1
DO 301 J=1,81
301 GRID(I,J)=BLANK
300 CONTINUE
XTTEST=XMAX*XMIN
YTTEST=YMAX*YMIN
IF (YTTEST)1,222,222
1 IF (XTTEST)3,33,444
3 IF (XTTEST)333*(-YMIN)/YRANGE+1.5
4 DO 40 I=1,IRYP1
40 GRID(I,IYAXIS)=SLINE
5 GOTO 222
IYAXIS=RY *XMAX/XRANGE+1.5
333 DO 60 I=1,81
60 GRID(IYAXIS,I)=DASH

```

C PLACING POINTS IN THEIR PROPER GRID POSITIONS

```

444 IF (MODCUR.EQ.0.OR.MODCUR.EQ.1) JSET=0
JSET=JSET+1 JSET=1
IF (JSET.GT.4) JSET=1
DO 700 I=1,NDATA,KK2
IPTY=RY *(X(I)-XMIN)/X RANGE+1.5
IPTX=80.*((Y(I)-YMIN)/YRANGE+1.5
IF (IPTY.GT.IRYP1.OR.IPTX.GT.81) GO TO 70
IF (IPTY.LE.0.OR.IPTX.LE.0) GO TO 70
GRID(IPTY,IPTX) = YCHAR(JSET)
GO TO 700
70 IERR=IERR+1
700 CONTINUE

```



```

C COMPUTE PROPER SCALE NUMBERS
C
IF(MODCUR.EQ.0.OR.MODCUR.EQ.1) GO TO 8000
IF(MODCUR.EQ.2) RETURN
GO TO 922
8000 YINCR=YRANGE/4*
YSCALE(1)=YMAX
YSCALE(1)=YMIN
DO 80 I=2,5
YSCALE(I)=YSCALE(I-1)-YINCR
DO 81 I=2,1XSC
XSCALE(I)=XSCALE(I-1)+XINCR
C OUTPUT SECTION WITH GRAPH
C
IF(MODCUR.EQ.0.OR.MODCUR.EQ.3)GO TO 922
RETURN
17 FORMAT(12X,
1P,E10.3,'4(10X,E10.3)' / 15X,2H***(3)
922 WRITE(6,17) YSCALE(5),YSCALE(4),YSCALE(3)
I=1
I=0
DO 101 IK=1,IRYP1
IF(I)91,91,92
WRITE(6,18) XSCALE(II),(GRID(IK,IY),IY=1,
FORMAT(3X,1P
1 II=II+1
I=I+1
IF(I-10)101,103,103
103 I=0
CONTINUE
101 WRITE(6,19) (GRID(IK,IY),IY=1,81)
FORMAT(15X,1H#,1X,81A1,1X,1H#)
102 I=I+1
IF(I-10)101,103,103
103 I=0
CONTINUE
101 WRITE(6,22) YSCALE(5),YSCALE(4),YSCALE(3)
FORMAT(15X,2H**,8(10H+*****),3H+***(3)
1001 IF(IERR) 1000,1000,1001
1001 WRITE(6,20) IERR
20 FORMAT(10X,NUMBER OF POINTS OUT OF RANG
C 1000 RETURN
END

```



```

000000866
000000877
000000888
000000899
000000900
000000910
000000920
000000930
000000940
000000950
000000960
000000970
000000980
000000990
000001000
000001010
000001020
000001030
000001040
000001050
000001060
000001070
000001080
000001090
000001100
000001110
000001120
000001130
000001140
000001150
000001160
000001170
000001180
000001190
000001200
000001210
000001220
000001230
000001240
000001250
000001260
000001270
000001280
000001290

C
WRITE(NUNIT,5026) { LABEL(JA), JA = 1,18 }
C   TITLE(JA), JA = 1,18 }
C   YNAME(JA,1), JA = 1,18 }
C   YNAME(JA,2), JA = 1,18 }
C   YNAME(JA,3), JA = 1,18 }
C   YNAME(JA,4), JA = 1,18 }
C   YNAME(JA,5), JA = 1,18 }
C   YNAME(JA,6), JA = 1,18 }
C   YNAME(JA,7), JA = 1,18 }
C   YNAME(JA,8), JA = 1,18 }
C   YNAME(JA,9), JA = 1,18 }
C   YNAME(JA,10), JA = 1,18 }
C   YNAME(JA,11), JA = 1,18 }
C   YNAME(JA,12), JA = 1,18 }
C   YNAME(JA,13), JA = 1,18 }

C   IF( AGTOUT .EQ. YES ) WRITE(NUNIT,5020)
C   IF( TAPE .EQ. YES ) WRITE(NUNIT,5021)
C   IF( CARDS .EQ. YES ) WRITE(NUNIT,5022)
C   IF( CARDS .EQ. YES ) WRITE(NUNIT,5023)
C   IF( DAMPNG .EQ. NO ) WRITE(NUNIT,5025)
C   IF( DAMPNG .EQ. YES ) WRITE(NUNIT,5024)
C   IF( IDRWA .EQ. YES ) WRITE(NUNIT,5047)
C   IF( IDRWA .EQ. NO ) WRITE(NUNIT,5048)
C   IF(CLEAR .LE. -10.0) WRITE(NUNIT,5042)
C   IF(CLEAR .GT. -10.0) WRITE(NUNIT,5041) CLEAR
C   IF(PIN2$8 .EQ. YES) WRITE(NUNIT,5003)
C   IF(PIN2$8 .EQ. NO) WRITE(NUNIT,5004)

C   WRITE(NUNIT,5010) RPM,COFD , CLEAR
C   WRITE(NUNIT,5017) XZERO,FZERO
C   IF(IMPRINT .EQ. NO) RETURN

C   WRITE(NUNIT,5027)
DO 80 JA=1,NC (CONFIG(JA,JB), JB = 1, NCLEAR)
WRITE(NUNIT,5028)

C

```



```

80 CONTINUE
C   WRITE(NUUNIT,5050)
DO 700 JA = 1, KSYS
      WRITE(NUUNIT,5051) LAYER(JA)
CONTINUE
C   WRITE(NUUNIT,5002)
WRITE(NUUNIT,5052)
DO 710 JA = 1, KSYS
      WRITE(NUUNIT,5053) (IDP(JA,JB), JB = 1, NCLEAR)
CONTINUE
C   WRITE(NUUNIT,5054)
DO 720 JA = 1, KSYS
      WRITE(NUUNIT,5055) (IQ(JA,JB), JB = 1, KSYS)
CONTINUE
C   WRITE(NUUNIT,5002)
WRITE(NUUNIT,5056)
DO 730 JA = 1, KSYS
      WRITE(NUUNIT,5057) IDIM(JA)
CONTINUE
C   WRITE(NUUNIT,5058)
DO 740 JA = 1, NCLEAR
      WRITE(NUUNIT,5053) (ICORD(JA,JB), JB = 1,2)
CONTINUE
C   WRITE(NUUNIT,5001)
C   WRITE(NUUNIT,5045)
C   DO 221 JA = 1, KSYSM1
      WRITE(NUUNIT,5044) (STFI(JA,JB), JB = 1,KSYSM1)
CONTINUE
C   WRITE(NUUNIT,5046)
C   DO 7 JA = 1, KSYSM2
      WRITE(NUUNIT,5044) (STFRI(JA,JB), JB = 1,KSYSM2)
CONTINUE
C   7
C   358 CONTINUE
C   WRITE(NUUNIT,620)
C   DO 640 JA = 1, ISYS

```



```

00001780
00001790
00001800
00001810
00001820
00001830
00001840
00001850
00001860
00001870
00001880
00001890
00001900
00001910
00001920
00001930
00001940
00001950
00001960
00001970
00001980
00001990
00002000
00002010
00002020
00002030
00002040
00002050
00002060
00002070
00002080
00002090
00002100
00002110
00002120

640 WRITE(NUNIT,1030) ( PHI(JA,JB), JB = 1, ISYS)
C   CONTINUE
C   WRITE(NUNIT,660)
C   WRITE(NUNIT,1030) ( ELAM(JA), JA = 1, ISYS)

C   359 CONTINUE
C   IF(MPRINT .EQ. NO) RETURN
C   WRITE(6,1049)
2   WRITE(NUNIT,1050) C,ISYS,ISYS

C   WRITE(NUNIT,1020)
DO 40 JA=1,ISYS
C   WRITE(NUNIT,1030) ( BMIV(JA,JB), JB = 1, ISYS)
C   WRITE(NUNIT,5002)
WRITE(NUNIT,1040)
DO 50 JA=1,ISYS
C   WRITE(NUNIT,1030) ( DA(JA,JB), JB = 1, ISYS)
C   CONTINUE
C   WRITE(NUNIT,5002)
WRITE(NUNIT,1060)
DO 60 JA=1,ISYS
C   WRITE(NUNIT,1030) ( DV(JA,JB), JB = 1, ISYS)
C   CONTINUE
C   70 CONTINUE
C   RETURN
END

```



```

C      DO 3 JA = 1,ISYS ( SYSM(JA,JB) , JB = 1,ISYS)
C      WRITE(NUNIT,1071)
C      DO 4 JA = 1,ISYS
C      WRITE(NUNIT,1030) ( DAMP(JA,JB) , JB = 1,ISYS)
C      CONTINUE
C      WRITE(NUNIT,1072)
C      DO 5 JA = 1,ISYS
C      WRITE(NUNIT,1030) ( STF(JA,JB) , JB = 1,ISYS)
C      CONTINUE
C      WRITE(6,1049)
C      WRITE(6,1048)
C      WRITE(NUNIT,1051) C,ISYS
C      WRITE(NUNIT,1073)
C      DO 6 JA = 1,ISYS
C      WRITE(NUNIT,1031) COLM(JA)
C      WRITE(NUNIT,1032) COLC(JA)
C      WRITE(NUNIT,1033) COLK(JA)
C      CONTINUE
C      RETURN
C      END

```



```

SUBROUTINE REDUCE( SYSM, DAMP, STF, KSYS )
C IMPLICIT REAL*8(A-H,O-Z)
C DIMENSION SYSM(KSYS,KSYS,1), DAMP(KSYS,KSYS,1), STF(KSYS,KSYS,1)
C
DO 10 JA =1, KSYS
DO 10 JB =1, KSYS
SYSM(JA,JB,2) = SYSM(JA,JB,1)
DAMP(JA,JB,3) = DAMP(JA,JB,1)
STF(JA,JB,3) = STF(JA,JB,1)
10 CONTINUE
C
DO 20 JC = 1, KSYS
SYSM(2,JC,3) = SYSM(2,JC,3) + SYSM(8,JC,3)
DAMP(2,JC,3) = DAMP(2,JC,3) + DAMP(8,JC,3)
STF(2,JC,3) = STF(2,JC,3) + STF(8,JC,3)
20 CONTINUE
C
DO 30 JD = 1, KSYS
SYSM(JD,2,3) = SYSM(JD,2,3) + SYSM(JD,8,3)
DAMP(JD,2,3) = DAMP(JD,2,3) + DAMP(JD,8,3)
STF(JD,2,3) = STF(JD,2,3) + STF(JD,8,3)
30 CONTINUE
C
      RETURN
END

```



```

00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
00000180
00000190
00000200
00000210
00000220
00000230
00000240
00000250
00000260
00000270
00000280
00000290
00000300
00000310
00000320
00000330
00000340
00000350
00000360
00000370

SUBROUTINE SHIFT( C, J, CONFIG, NPRINT)
C
C   INTEGER CSAV, CSAVE, C, CONFIG, BINARY, YES
C
C   DIMENSION CONFIG(16,1)
C
C   COMMON NUNIT, IPRINT
C
C   DATA YES/4HYES /, NO/4HNNO /
C
C   BINARY(I3,I2,I1) = I3*4 + I2*2 + I1
C
C   10 FORMAT('0',I1,I1,I1) 10X, *FROM SHIFT, C=' , I1,*
C   11 FORMAT('0',I1,I1,I1) 10X, *NEW C = ' , I1)
C
C   IF(NPRINT .EQ. NO) GO TO 12
C
C   WRITE(NUNIT,10) C,C,J,CONFIG(C,J)
C
C   12 CONSAV = CONFIG( C, J )
C   CSAVE = C
C   CONFIG(C,J) = CONSAV + 1
C
C   IF( CONFIG(C,J) .EQ. 2 ) CONFIG(C,J) = 0
C
C   C = BINARY( CONFIG(C,1), CONFIG(C,2), CONFIG(C,3) ) + 1
C
C   IF(NPRINT .EQ. NO) GO TO 20
C
C   WRITE(NUNIT,I1) C
C
C   20 CONFIG(CSAVE,J) = CONSAV
C
C
C   RETURN
END

```



```

C 5000 FORMAT('0','CONFIGURATION =',I3,5X,'DEGREES =',F9.3,10X,'WT =',
      1      1PD15.7)
C 5001 FORMAT('1')
C 5002 FORMAT('0','BACKSTEP ROUTINE CALLED DUE TO CHANGE IN ',
      1      'CONFIGURATIONS')
C 5003 FORMAT('0','RETURNED TO MAIN FROM BACKSTEP ROUTINE')
C 5004 FORMAT('0','IN MAIN - LOCATION ',I4,' -- CSAVE =', I2,
      1      ' AND C =', I2)
C 5005 FORMAT('0','W = ',I1PD15.7,' AMPW = ',D15.7,' W2 = ', D15.7,
      1      'AMPW2 = ',D15.7)
C 5007 FORMAT('0','VALUES CALCULATED AFTER BACKSTEP TAKEN :')
C 5015 FORMAT(5E10.4,2A4)
C 5016 FORMAT('0','IMPACT FORCE AT COORDINATE ',I2,' =',1PD15.7,
      1      ' - DEGREES ',OPF9.3)
C 5026 FORMAT('0',10X,20A4)
C 5038 FORMAT(20A4)
C 5039 FORMAT('0',20X,20A4)
C 5040 FORMAT('0',30X,20A4)
C 5041 FORMAT('0',5X,15$('0X,1PE10.3') )
C 5050 FORMAT('0',40X,20A4)
C 5051 FORMAT('0',50X,20A4)
C 5062 FORMAT('0',23X,'DEGREES =',F9.3,10X,'WT =',1PD15.7)
C 5063 FORMAT('0',23X,'CAMQ =',1PD18.8,'10X,'CAMQDD =',D18.8,'10X,'CAMQDD =',
      1      1PD18.8)
C 5310 FORMAT('0',5(1PD15.7, 5X) )
C
C **** READ INPUT VARIABLES ****
C
C 50 CALL INPUT(LABEL1, TITLE, XNAME, YNAME,
      1      AGTOUT, CARDS, TAPE, IPLOT, IDRAW,
      2      KPRINT, DAMPNG, RPM, CLEAR, WTDEG, AMP, DELT,
      3      ROCKER, AXILTH, NTIME, KSYS, NCLEAR, NC,
      4      REALTH, NCOUNT, IDEG, PIN2$8, MPRINT)
      5
C
C FS = +145.000
C PI = 3.14159265359000
C DEGR = 0.
C WT = 0.
C NDEG = 0
C NDEGKP = 0

```



```

ICOUNT = 0
KACT = 0
IDT = 0
IMP = 0.000
COFD = 1.000
IUNIT = 6
ICORD(1,1) = 1
ICORD(1,2) = 9
ICORD(2,1) = 2
ICORD(2,2) = 8
ICORD(3,1) = 7
ICORD(3,2) = 10
DT = DELT
NPRINT = IPRINT
SEAT = YES
C
NSYS = KSYS + 2
DEGOP1 = 180.0D0/PI
KSYSM1 = KSYS-1
KSYSM2 = KSYS-2
KSYSM3 = KSYS - 3
NCDFN = NC + 1
NCP1 = NC + 1
C
Q(10) = CLEAR
QDD(10) = 0.0D0
QDSAV(10) = CLEAR
QDDSAV(10) = 0.0D0
QDDSAV(10) = 0.0D0
W = RPM*PI/30.0D0
AMPW = AMP*W
W2 = -W*W
AMPW2 = -AMPW*W
DTW = DT*W
WTSTOP = WTDEG
WTPRNT = WTDEG
+ 15.0D0*DTW*DEGOP1
- 3.0D0*WTDEG*DEGOP1
C
WRITE(NUNIT,5005) W1,AMPW,W2,AMPW2
WRITE(NUNIT,5310) WTSTOP,WTPRNT,WTDEG, DTW, DEGOP1
WRITE(NUNIT,5039) (LABEL(JA),JA=1,18)
WRITE(NUNIT,5039) (TITLE(JA),JA=1,18)
C
IF(TAPE .EQ. YES) IUNIT = 4
IF(CARDS .EQ. YES) IUNIT = 7
IF(CLEAR .LE. -10.0) SEAT = NO
00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030
00001040
00001050
00001060
00001070
00001080
00001090
00001100
00001110
00001120
00001130
00001140
00001150
00001160
00001170
00001180
00001190
00001200
00001210
00001220
00001230
00001240
00001250
00001260
00001270
00001280
00001290
00001300
00001310
00001320
00001330
00001340
00001350
00001360
00001370
00001380
00001390
00001400
00001410
00001420

```





```

C   C 180 CONTINUE
C   C WRITE(NUNIT,5001)
C   C INITIAL CONTACT AT CAM AND ROCKER
C   C      C=7      CSAVE = C + 1
C   C
C   C***** CONVERT TO REAL * 4
C   C***** OUTPUT FOR AGT
C   C***** *****
C   C
C   IF(AGTOUT .EQ. NO ) GO TO 265
C   RXZERO = XZERO
C   RAMP = AMP
C   RRPM = RPM
C   RCLEAR = CLEAR
C   WNR4 = WN
C
C   IF(IUNIT .EQ. 6) GO TO 260
C   WRITE(IUNIT, 5038) {LABEL(JA),JA=1,18}
C   WRITE(IUNIT, 5038) {TITLE(JA),JA=1,18}
C
C   260 WRITE(IUNIT,5015) RXZERO,RAMP,RRPM, WNR4, RCLEAR, SEAT, PIN2$8
C   265 CONTINUE
C
C   ***** CONFIGURATION CALCULATIONS *****
C
C   ***** INITIALIZING ROUTINE *****
C
C   360 WT = WT + DTW
C
C   DO 361 JA = 1, NSYS
C   CALL ADJUST(JA,JA,QSAV,QDDSAV,Q, QD, QDD)
C   361 CONTINUE
C
C   DO 364 I = 1, KSYS

```



```

FSAV(I) = F(I)
FSAV(I) = DABS(F(I))
CONTINUE
C   IF(DEGR .GT. WTPRNT) NPRINT = YES
C   IF(DEGR .GT. WTSTOP) STOP
C   CALL CAM(WT,AMP,AMPW,XZERO,W2,Q,QQ,QDD)
C   IF(C .NE. CSAVE .AND. NPRINT .EQ. YES) WRITE(UNIT,5000) C,DEGR,WT
C   IF(C .EQ. CSAVE .AND. NPRINT .EQ. YES) WRITE(UNIT,5062) DEGR,WT
C
C   CALL BRANCH(C,F,KSYS,ISYS,Q,QQ,QQDSAV,ICORD,DT,NPRINT,
C               1,IMP,PIN2$8, IDIM)
C
C   390 CALL NEWMRK( ISYS, KSYS, C,D, STF, DAMP, SYSM, DP,
C   1 DT, DTO2, DT203, Q, QD, BMIN(1,1,C), DA(1,1,C),
C   2 DV(1,1,C), NC, ALFA, BETA, CONFIG(1,2), F, ICORD, NCLEAR, NPRINT, IQ,
C   3 COLM, COLC, COLK, IDP, PIN2$8, IDIM, LAYER, IGU, QDDSAV,
C   4 QDSAV, QSAV, SCALE, IMP)
C
C   IF(((KPRINT .EQ. YES) .OR. (NPRINT .EQ. YES)) .AND. (IMP .NE. 0))
C   1 WRITE(UNIT,5016) IMP, F(IMP), DEGR
C
C   IF(NPRINT .EQ. YES) CALL DEBUG( 2, Q, QDD, F, KSYS, ISYS, DP,
C   1, NDEG, NPRINT, DEGR, C)
C
C   400 CSAVE = C
C   405 IF(NDEG .LT. IDEG) GO TO 463
C
C   *****
C   **** CHECK TO SEE IF
C   **** MONITORED VALUES HAVE BEEN
C   **** EXCEEDED AND IF SO SHIFT TO THE
C   **** APPROPRIATE CONFIGURATION
C   **** *****
C
C   410 CALL MONTOR(NCLEAR,CONFIG(1,2),ICORD, F, C, NPRINT, IDT, DTP,
C   DTPM, IMPACT, Q, QSAV, DT)

```



```

C **** DETERMINE IF IMPACT AND AT WHAT COORD. ***
C **** CONTINUE IMPACT(IMP) ) GO TO 418
C
C DO 415 IMP = 1 NCLEAR
C IF(IIDT .EQ. IMPACT(IMP) ) GO TO 418
C
C 415 CONTINUE
C IMP = 0
C GO TO 420
C
C 418 IMP = ICORD(IMP,1)
C 420 IF(CSAVE .EQ. C) GO TO 450
C
C IF(NPRINT .EQ. YES) WRITE(NUNIT,5002)
C
C 440 CALL BKSTEP(BMIV(1,1,NCP1), DA(1,1,NCP1), DV(1,1,NCP1),
C 1      SYSM, DAMP, STF, CSAVE, KPRINT, KSYS, WT,
C 2      DP, Q, QDD, BETA, CONFIG(1,2),
C 3      ICORD, QD, ALFA, XZERO, DTPM, ISYS, AMPW,
C 4      ICORD, NCLEAR, AMP, XZERO, DTPM, PIN2$8,
C 5      W2, KSYSMI, KSYSM2, DTW, NSYS, IDT, IDIM,
C 6      COLM, COLK, IDP, QSAV, QDDSAV, FSAV,
C      LAYER, C, IGU, SCALE)
C
C IF(NPRINT .EQ. YES) WRITE(NUNIT,5003)
C
C **** PROGRAM PARAMETERS ARE NOW CALCULATED ***
C **** ESTABLISH PLOT PACKAGE ***
C ****
C
C 450 CALL FIXFOR(CSAVE, F)
C GO TO 465
C
C 463 DO 464 I = 1,KSYS
C 464 FSAV(I) = DABSS(F(I))
C
C 465 IF(DEGR .EQ. 0.0) CALL DEBUG(2,Q, QDD, QDD, F, KSYS, ISYS, DP, NDEG,
C 1
C DEGR = WT*DEGOPI
C NDEGKP = NDEG
C NDEG = DEGR
C
C IF( NDEG .EQ. NDEGKP ) GO TO 360

```



```

533 DEG( NDEG ) = DEGR
      INTIME = NDEG + NTIME
      DEG( INTIME ) = DEGR

      IF( AGTOUT .EQ. NO ) GO TO 535
      IF( NDEG .LT. 360 .OR. NDEG .GT. 725 ) GO TO 535
      IF( IAGT .LT. 2 ) GO TO 535
      IAGT = 0
      KAGT = KAGT + 1

C     CALL AGT( Q, QR4, WT, WTR4, XZERO, ROCKER, NPRINT, AXILTH,
      2      R'AXILTH, C, IUNIT, KAGT)
      535 CONTINUE

C     CALL ARRAY( F, Q, QD, QDD, NDEG, FPLOT, NTIME)
      C
      IF( KPRINT .EQ. NO ) GO TO 634

C     ICOUNT = ICOOUNT + 1
      IF( ICOOUNT .LT. NCOUNT ) GO TO 634
      ICOOUNT = 0

C     CALL DEBUG( C, Q, QD, QDD, F, KSYS, ISYS, DP, NDEG, NPRINT, DEGR,
      1      NCOUNT )
      1 CONTINUE

C     636 IF( NDEG .LT. NTIME) GO TO 360
      C
      **** PLOT PACKAGE FOR SPECIFIED GRAPHS ****
      C
      **** **** **** **** **** **** **** **** ****
      C
      CALL GRAPH( FPLOT, YNAME, XNAME, DEG1, DEG2, Y1, Y2, Y3, Y4,
      1      YNAME1, YNAME2, PIN2$8, CLEAR,
      2      NTIME, LABEL, TITLE, IDRAW, IPLOT, XSCALE,
      3      IP, NPRINT, GRID, LINES)

C     GO TO 50
      END
      4037 RETURN

```


SDS-9300 COMPUTER PROGRAM LISTING

This listing is coded in SDS FORTRAN IV, (Ref. 5-7), to run on a Scientific Data System Model 9300 computer interfaced with an ADAGE Graphics Display Terminal, Model 10 (AGT-10). Due to the limited core storage of this computer, overlaying techniques have been employed.


```

SENSE SWITCH 1 -- RESTART - READ
SENSE SWITCH 2 -- INPUT(101)
SENSE SWITCH 3 -- CALL CAM
SENSE SWITCH 4 -- CALL JOINT
SENSE SWITCH 5 -- IF MOD EACH CYCLE
SENSE SWITCH 6 -- PIN28
SENSE SWYES, RLNK, CONFIG
SENSE ROTATE,Y(107),IRRAY2(18),IRRAY3(13),IRRAY4(32),
SENSE ON IRRAY1(24),IRRAY6(36),IRRAY7(24),ISPRNG(9),IRRAY9(49),
SENSE DIMENSION IRRAY5(49),IRRAY10(49),1101(7),1105(3),1104(8),1117(5),
SENSE DIMENSION IRAY10(12),1107(11),1108(7),1109(7),1110(2),1115(1),1116(10),
SENSE DIMENSION IRAY11(5),1112(9),1113(8),1114(11),1115(1),1116(10),
SENSE DIMENSION X(9),Y(9),CONFIG(180),YZERO(9),
SENSE DIMENSION RADCOS(97),RADSIM(97),
SENSE DIMENSION AXLTH1,AXLTH2,AXLTH3,AXLTH4,
COMMON WTH1,WTH2,WTH3,WTH4
COMMON PI,RAD,DEG,IDEV
COMMON IGDIR(20),ITDIR(40)
COMMON XA,YA,XB,YB,XC,YC,XD,YD,XE,YE,XF,YF,XG,YG,XH,YH,XS,YSS,XP,YP
COMMON Q1(180),Q2(180),Q3(180),Q4(180),Q5(180),Q6(180),Q7(180)
COMMON Q8(180),Q9(180),CMEGA(180)
COMMON XCAM,YCAM
COMMON DATA NULL/77777777B/, BLNK/4H /, YES/4HNO /, NO/4HNO /
FORMAT(1PE12.2,28X) 102
FORMAT(1FREQ,FATI0 103
FORMAT(1CONTACT,*) 104
FORMAT(1CAMROCK,*) 105
FORMAT(1NOCONTACT,*) 106
FORMAT(1VALVESEAT,*) 107
FORMAT(1CAMROTAT,*) 108
FORMAT(1ROCKER,*) 109
FORMAT(1CAM,*) 110
FORMAT(1CAMROTAT,*) 111
FORMAT(1CAM,*) 112
FORMAT(1CAMROTAT,*) 113
FORMAT(1CAM,*) 114
FORMAT(1CAMROTAT,*) 115
FORMAT(1CAM,*) 116
FORMAT(1CAMROTAT,*) 117
FORMAT(1CAM,*) 118
FORMAT(1CAMROTAT,*) 119
FORMAT(1CAM,*) 120
FORMAT(1CAMROTAT,*) 121
FORMAT(1CAM,*) 122
FORMAT(1CAMROTAT,*) 123
FORMAT(1CAM,*) 124
FORMAT(1CAMROTAT,*) 125
FORMAT(1CAM,*) 126
FORMAT(1CAMROTAT,*) 127
FORMAT(1CAM,*) 128
FORMAT(1CAMROTAT,*) 129
FORMAT(1CAM,*) 130
FORMAT(1CAMROTAT,*) 131
FORMAT(1CAM,*) 132
FORMAT(1CAMROTAT,*) 133
FORMAT(1CAM,*) 134
FORMAT(1CAMROTAT,*) 135
FORMAT(1CAM,*) 136
FORMAT(1CAMROTAT,*) 137
FORMAT(1CAM,*) 138
FORMAT(1CAMROTAT,*) 139
FORMAT(1CAM,*) 140
FORMAT(1CAMROTAT,*) 141
FORMAT(1CAM,*) 142
FORMAT(1CAMROTAT,*) 143
FORMAT(1CAM,*) 144
FORMAT(1CAMROTAT,*) 145
FORMAT(1CAM,*) 146
FORMAT(1CAMROTAT,*) 147
FORMAT(1CAM,*) 148
FORMAT(1CAMROTAT,*) 149
FORMAT(1CAM,*) 150
NAMELIST CONFIG, ISEAT, SF1, SF2, SF3, SF4, SF1A, SF2A, SF3A, SF4A, WN, TH1, TH2, AXLT, H3, H4, FREQ, DTHETA, AXLT, H2, AXLT, H3, AXLT, H4, IDEV, XA, XB, XC, XD, XE, XF, XA, XB, XC, XD, XE, XF, XG, YG, YE, YD, YH, YS, YP, YCAM, DIFF, CTOGD(A) = A / 0.2000 * A + 5.000, GTOCM(A) = 0.2000 * A - 1.0000
IDEV = 1
CONTINUE
PI = 3.14159265
TWOPI = 2.0*PI
NTIME = 180
ISEAT = YES
DEG = 180.0/PI

```

CCCCC


```

RAD = PI/180.0
SF1 = 1.0
SF2 = 1.0
SF3 = 1.0
SF4 = 1.0
SF1A = 15.0
SF4A = 15.0
WTH1=WTH2=WTH3=WTH4=0.4
AXLTH1 = 7.9
AXLTH2 = 3.4
AXLTH3 = 5.1
AXLTH4 = 3.3
XXA = 0.9
XXB = 3.9
XXC = 10.2
XXD = 0.5
XXE = 10.0
XXF = 0.5
XXG = 3.9
XXH = 9.0
XP = 11.2
XS = 16.8
CALL INITIAL(CONFIG, IRRAY1, IRRAY2, IRRAY3, IRRAY4, IRRAY5, IRRAY6, IRRAY7, IRRAY9,
1          IRRAY10, ISPRING)
2          CALL READ(NTIME, TIME, I104, I105, XZERO, AMP, RPM, WN, ISEAT, PIN28,
1          CONFIG, DIFF)
1          THETA = 0.0
RADIUS = 1.5
DTHETA = 4.0*RAD
IF(SENSE_SWITCH) INPUT 21260
2          CONTINUE
OUTPUT(101) * SPRCTR CALLED *
X           1260
          OUTPUT(6) * SPRCTR CALLED *
          CALL SPRCTR(IRRAY3, ISEAT, DIFF)
          CALL MIDDBAR(IRRAY2, IRRAY9)
DO 1400 I = 6, 97
        THETA = THETA + DTHETA
        RADCOS(I) = RADIUS*COS(THETA)
        RADSIN(I) = RADIUS*SIN(THETA)
CONTINUE
1400      AMPO2 = AMP/2.0
        FREQ = (RPM/2.*PI) / (WN * 30.0 )
ENCODE(40,102,1116) FREQ
ENCODE(32,103,1113)
ENCODE(20,1450,1111)
ENCODE(8,1455,1110)
ENCODE(28,1430,1109)

```



```

ENCODE( 32, 1431, 1104)
ENCODE( 28, 1432, 1101)
ENCODE( 12, 1433, 1105)
ENCODE( 20, 1434, 1117)
ENCODE( 28, 1460, 1108)    7 24,30,2,3,IER)
CALL(TEXTO(IDEV1PUT(101) IER)
IF(IER•NE•0) OUTPUT(101) IER
N= 0 ; I = 0 ; K = 1, NTIME
DO 1500 IF( K .EQ. NTIME ) OUTPUT(6) ! COMPLETE CYCLE!
N= N+1 CALL DISPLAY(CONFIG, K, 1105, 1109, 1117, 1101, 1104)
X 3
      1551
      CALL SCAM( 97, IRRAY1, RADIUS, AMP, I, RADCOS, RADSIN)
      ROTATE = IFIX( OMEGA(I)* DEG )
      ENCODE(410, IDEV1PUT(1115) ROTATE
      CALL(TEXTO(IDEV1PUT(101) IER)
      IF(IER•NE•0) OUTPUT(101) IER
      CALL(TEXTO(IDEV1PUT(101) IER)
      IF(IER•NE•0) OUTPUT(101) IER
      CALL(BAR1, I, IRRAY4, SF1, SF1A, PIN28)
      CALL(BAR2, I, IRRAY5, SF2, SF2, IRRAY9, PIN28)
      CALL(BAR3, I, IRRAY6, SF3, SF3, IRRAY10)
      CALL(BAR4, I, IRRAY7, SF4, SF4A)
      IF( I •NE• 1 ) GO TO 175
      X(1) = Y(1) = 0.0
      X(2) = XC ; Y(2) = YC
      X(3) = X(5) = X(7) = XH - 0.6
      X(4) = X(6) = XH + 0.6
      X(8) = XH
      DO 50 Y(J+2) = FLOAT(J)*(YH-YC)/6.0 + YC
      Y(8) = YH
      X(9) = Y(9) = 0.0
      ISPRNG(1) = THEAD(0,10)
      DO 100 J = 1, 9
      YZERO(J) = Y(J)
      50 CONTINUE
      50

```



```

00001450
00001460
00001470
00001480
00001490
00001500
00001510
00001520
00001530
00001540
00001550
00001560
00001570
00001580
00001590
00001600
00001610
00001620
00001630
00001640
00001650
00001660
00001670
00001680
00001690
00001700
00001710
00001720
00001730
00001740
00001750
00001760

X(J) = GDTOCM( X(J) ) ; Y(J) = GDTOCM( Y(J) )
100  CONTINUE
ISPRNG(2) = IPACK( X(2), Y(2), 0 )
      DD150 J = 38
      DO ISPRNG(J) = IPACK( X(J), Y(J), 1 )
150  CONTINUE
ISPRNG(9) = 0
      CALL GRAPHO( IDEV, ISPRNG, 9, 8, IER )
      IF( IER .NE. 0 ) OUTPUT(101) IER
      IF( I .EQ. 1 ) GO TO 1990
175  CONTINUE
DO 200 J = 3, 8
      Y(J) = CMTOGD( Y(J) )
      Y(J) = YZERO( J ) + FLOAT( J-2 ) * Q4( 1 ) * SF4 / 6.0
      Y(J) = GDTOCM( Y(J) )
      ISPRNG(J) = IPACK( X(J), Y(J), 1 )
200  CONTINUE
      CALL GRAPHO( IDEV, ISPRNG, 9, 8, IER )
      IF( IER .NE. 0 ) OUTPUT(101) IER
1990  CONTINUE SWITCH 1 ) 1, 1991
1991  CONTINUE
      IF( SENSE SWITCH 2 ) 2, 1992
1992  CONTINUE
      IF( SENSE SWITCH 6 ) 6, 2000
      6   CALL TEXTR( IDEV, NUL1, 3946, 2, 3, IER )
      IF( IER .EQ. 0 ) OUTPUT(101) IER
1998  IF( MUD( ITDIR(1), 8 ).EQ.0 ) GO TO 1998
2000  CONTINUE
      I = 1 ; N = 1
      GO TO 1500
END

```



```

SUBROUTINE BAR1(JA,LINK1,SF1A,PIN28)
INTEGER PIN28, YES
DIMENSION LINK1(32), X(31), Y(31), AMP(8)
COMMON AXLTH1, AXLTH2, AXLTTH3, AXLTTH4,
COMMON WTH1, WTH2, WTH3, WTH4
COMMON PI, RAD, DEG, IDEV
COMMON IGDIR(20), ITDIR(40)
COMMON YA, XB, YB, XC, XD, YE, XF, YC, XH, YH, XS, XP, YP
COMMON Q1(180), Q2(180), Q3(180), Q4(180), Q5(180), Q6(180), Q7(180)
COMMON Q8(180), Q9(180), QMEA(180)
COMMON XCAM, YCAM
DATA YES/4 HYES /, NO/4 HNO /
CMTOGD(A) = A / 2000 * A + 5.000
GDTOCM(A) = 0.2000 * A - 1.0000
NAMELIST IER
LINK1(1) = IHEAD(0,10)
IF( JA .NE. 1 ) GOTO 500
X(2) = XD ; Y(2) = YD
X(3) = Y(30) = Y(2) = Y(2)
Y(1) = Y(33-I) = Y(3) - 0.2
DO 30 I = 1, 33-I
X(I) = X(2) - WTH1/2.0
X(I-1) = X(3) + WTH1
CONTINUE
X(15) = X(16) = X(2) - 3*WTH1
X(17) = X(18) = X(2) + 3*WTH1
X(19) = X(19)
Y(15) = Y(18) = Y(19) = Y(14)
Y(16) = Y(17) = Y(14) - WTH1
YZER14 = Y(14)
DO 45 I = 1, 31
X(I) = GDTOCM(X(I)) ; Y(I) = GDTOCM(Y(I))
CONTINUE
45 DO 50 I = 2, 31
IF( I .EQ. 1 ) LINK1(2) = IPACK(X(1), Y(1), 1)
LINK1(I) = IPACK(X(I), Y(I), 1)
CONTINUE
50 LINK1(32) = 0
CALL GRAPHO( IDEV, LINK1, 32, 4, IER )
IF( IER .NE. 0 ) OUTPUT(101, IER, 'BAR1-' )
FORMAT( *, 3X, I=1, 13, 2E15.4 )
IF( JA .EQ. 1 ) RETURN
50 DO 550 I = 2, 31
X(I) = CMTOGD(X(I)) ; Y(I) = CMTOGD(Y(I))

```



```

550 CONTINUE
IF(PIN28.EQ.YES) Y(3)=Y(3)+0.2
Y(2)=Y(2)+0.2
Y(31)=Y(2)
Y(31)=Y(15)
Y(14)=Y(17)
Y(16)=Y(17)
DIFF=ABS(Y(3)-Y(14))-AXLTH1
DO 560 I=4,13
Y(I)=Y(33-I)-Y(3)-(FLOAT(I-3)*((Y(3)-Y(14))/11.0))
CONTINUE
DO 575 I=4,8
AMP(I)=DIFF*SIN(FLOAT(I-3)*PI/11.0)*SF1A
IF(AMP(I).GT.WTH1/2.0)AMP(I)=WTH1/2.0
FORMAT(,3X,,I3,AMP=,E15.5)
X(I)=X(I-1)=X(I+16)=X(I)+AMP(I)
X(33-I)=X(I+16)=X(I)+WTH1-2.0*AMP(I)
CONTINUE
DO 600 I=2,31
X(I)=GDTOCM(X(I));Y(I)=GDTOCM(Y(I))
LINK1(I)=IPACK(X(I),Y(I),1)
CONTINUE
LINK1(2)=IPACK(X(2),Y(2),0)
LINK1(32)=0
98 CALL GRAPHO(LDEV,LINK1,32,4,IER)
IF(IER.NE.0) OUTPUT(101)IER,BAR1-2,
RETURN
END

```



```

SUBROUTINE BAR2(JA,LINK2,SF2,IRRAY9,PIN28)
INTEGER PIN28,YES
DIMENSION LINK2(24),X(24),Y(24),PHI(10,3),YZERO(24)
COMMON AXLTH1,AXLTH2,AXLTH3,AXLTH4,WTH2,WTH3,WTH4
COMMON PI,RAD,DEG,IDEV
COMMON IGDIR(20),ITDIR(40)
COMMON XA,YA,XB,YB,XC,YC,XD,YD,XE,YE,XF,YF,XH,YH,XS,YS,XP,YP
COMMON Q1(180),Q2(180),Q3(180),Q4(180),Q5(180),Q6(180),Q7(180)
COMMON Q8(180),Q9(180),Q10(180),Q11(180),Q12(180),Q13(180),Q14(180)
COMMON XCAM,YCAM
DATA YES/4 HYES/NO/4 HNO/
GDTOCM(A)=0.2000*A-1.0000
CMTOGD(A)=A/0.2000+5.000
NAMELIST IER
L=1
K=2
IF(JA .NE. 1) GO TO 200
X(1)=Y(1)=0.0
X(24)=XF=Y(24)
FACT2=Y(24)-WTH2/2.0
FACT3=Y(24)+WTH2/2.0
DO 10 I=2,12
Y(1)=YZERO(I)=FACT2
Y(I+1)=FACT3
10 CONTINUE
DO 25 I=1,11
X(I)=X(I+1)=X(24) + FLOAT(I-1) * AXLTH2 / 10.0
X(13-I)=X(12+I)=X(24) + FLOAT(I-1) * AXLTH2 / 10.0
25 CONTINUE
XA=X(12);YA=Y(12)
IF(PIN28.EQ.NO) GO TO 26
IF(SENSE.SWITCH 5) 5,26
CALL JOINT(37,IRRAY9,0.1,10.0,K,JAI)
CONTINUE
AL22=AXLTH2**2
AL23=AXLTH2**3
DO 30 I=3,12
XX=X(I)-X(24)
XX2=XX**2
XX3=XX2*XX
PHI(I-2,1)= (AL23 - 3.0*XX2*AXLTH2 + 2.0*XX3 ) * SF2/AL23
PHI(I-2,3-L)= (XX*AL22 - 2.0*XX2*AXLTH2 + XX3 ) * SF2/AL22
PHI(I-2,4-L)= -(XX2*AXLTH2 - XX3 ) * SF2/AL22
30 CONTINUE
DO 50 I=1,23
FORMAT(*,3X,1,I=1,13,2E15.4)
50
99

```



```

X(I) = GDTOCM( X(I) ) ; Y(I) = GDTOCM( Y(I) )
50 CONTINUE
LINK2(1) = IHEAD(0,10)
LINK2(2) = IPACK(X(2), Y(2), 0)
DO 100 I = 3, 23
LINK2(I) = IPACK(X(I), Y(I), 1)
100 CONTINUE
LINK2(24) = 0
CALL GRAPHO( IDEV, LINK2, 24, 5, IER )
IF( IER .NE. 0 ) OUTPUT(101) IER, !BAR2-1
IF( JA .EQ. 1 ) RETURN
200 CONTINUE
DO 300 I = 3, 12
Y(I) = CMTOGD( Y(I) )
Y(25-I) = CMTOGD( Y(25-I) )
W = PHI(I-2,1)*Q2(JA) + PHI(I-2,3-L)*Q3(JA) + PHI(I-2,4-L)*Q6(JA)
Y(I) = W + YZERO(I)
Y(25-I) = WTH2 + Y(I)
300 CONTINUE
X(12) = CMTOGD( X(12) );
YA = Y(12)
XA = X(12)
IF( PIN28 .EQ. NO ) GO TO 326
IF( SENSE .SWITCH 5 ) 325,326
325 CALL JOINT( 37, IRAY9, 0.1, 10.0, K, JA )
326 CONTINUE
X(12) = GDTOCM( X(12) )
DO 350 I = 3, 12
Y(I) = GDTOCM( Y(I) )
Y(25-I) = GDTOCM( Y(25-I) )
350 CONTINUE
DO 400 I = 3, 23
LINK2(I) = IPACK(X(I), Y(I), 1)
400 CONTINUE
CALL GRAPHO( IDEV, LINK2, 24, 5, IER )
IF( IER .NE. 0 ) OUTPUT(101) IER, !BAR2-2
RETURN
END

```



```

SUBROUTINE PAR3(JA,LINK3,SF3,IRRAY10)
DIMENSION LINK3(36),X(35),Y(35),PHI(16,3),YZERO(35)
COMMON AXLTH1,AXLTH2,AXLTH3,AXLTH4,WTH1,WTH2,WTH3,WTH4
COMMON PI,RAD,DEG,IDEV
COMMON IGDIR(20),ITDIR(40)
COMMON XA,YA,XB,YB,XC,YC,XD,YD,XE,YE,XF,YF,XH,YH,XG,YG,XS,YS,XP,YP
COMMON Q1(180),Q2(180),Q3(180),Q4(180),Q5(180),Q6(180),Q7(180)
COMMON Q8(180),Q9(180),OMEGA(180)
COMMON XCAM,YCAM
GDTOCM(A)=0.2000*A-1.0000
CMTOGD(A)=A/0.2000-5.0000
NAMELIST IER
L=1
K=3
IF(JA .NE. 1) GO TO 200
X(1)=Y(1)
X(2)=XG
DO 10 I=2,18
Y(I)=YZERO(I)
Y(I+17)=Y(2)+WTH3
10 CONTINUE
DO 25 I=X(37-I),18
X(I)=X(2)+FLOAT(I-2)*AXLTH3 / 16.0
25 CONTINUE
XA=X(18);YA=Y(18)
IF(SENSE SWITCH 5) 5,26
CALL JOINT(37,IRRAY10,0.1,10.0,K,JA)
26 CONTINUE
AL32=AXLTH3**2
AL33=AXLTH3**3
DO 30 I=3,18
XX=X(I)-X(2)
XX2=XX**2
XX3=XX2*
PHI(I-2,2-L)=(+3.0*XX2*AXLTH3-2.0*XX3)*SF3/AL33
PHI(I-2,3-L)=( XX*AL32-2.0*XX2*AXLTH3+XX3)*SF3/AL32
PHI(I-2,4-L)=-( XX2*AXLTH3-XX3)*SF3/AL32
30 CONTINUE
DO 50 I=1,35
X(I)=GDTOCM(X(I));Y(I)=GDTOCM(Y(I))
50 CONTINUE
LINK3(1)=IHEAD(0,10)
LINK3(2)=IPACK(X(2),Y(2),0)
DO 100 I=3,35
LINK3(I)=IPACK(X(I),Y(I),1)

```



```

100 CONTINUE
LINK3(36) = 0
CALL GRAPHO(IDEV,LINK3(101)36,IER,BAR3-1),
IF(IER.NE.0) OUTPUT(101) IER, RETURN
IF(JA.EQ.1) RETURN
CONTINUE
DO 300 I = 3, 18
Y(1) = CMTOGD(Y(I))
Y(37-I) = CMTOGD(Y(37-I))
W = PHI(I-2,2-L)*Q4(JA) + PHI(I-2,3-L)*Q6(JA) + PHI(I-2,4-L)*Q5(JA)
W = W + YZERO(I)
Y(37-I) = WTH3 + Y(I)
CONTINUE
300 X(18) = CMTOGD(X(18))
XA = X(18); YA = Y(18)
IF(SENSE SWITCH 5) 325,326,
CALL JOIN(37,IRRAY10,0,1,10,0,K,JA)
CONTINUE
325 X(18) = GDTOCM(X(18))
DO 350 I = 3, 18
Y(1) = GDTOCM(Y(I))
Y(37-I) = GDTOCM(Y(37-I))
CONTINUE
350 DO 400 I = 3, 35
LINK3(1) = IPACK(X(I),Y(I),1)
CONTINUE
400 CALL GRAPHO(IDEV,LINK3(101)36,IER,BAR3-2)
IF(IER.NE.0) OUTPUT(101) IER, RETURN
END

```



```

SUBROUTINE BAR4( JA, LNK4, SF4A, SF4A )
DIMENSION LNK4( 24 ), X( 23 ), Y( 23 ), AMP( 7 )
COMMON AXLTH1, AXLTH2, AXLTH3, AXLTH4
COMMON WTH1, WTH2, WTH3, WTH4
COMMON PIGDIR( 20 ), ITDIR( 40 ), IDEV
COMMON XA, YA, XB, YB, XC, XD, YE, XF, YF, XS, YH, XP, YP
COMMON Q1( 180 ), Q2( 180 ), Q3( 180 ), Q4( 180 ), Q5( 180 ), Q6( 180 ), Q7( 180 )
COMMON Q8( 180 ), Q9( 180 ), QMEGA( 180 )
COMMON XCAM
GDTOCM( A ) = 0. * 2000 * A - 1.0000
CMTOGD( A ) = A / 0.2000 + 5.0000
NAMELIST IER
LNK4( 1 ) = IHEAD( 0, 10 )
IF( JA .NE. 1 ) GOTO 500
X( 1 ) = Y( 1 ) = 0.0
X( 2 ) = XH ; Y( 2 ) = YH
X( 23 ) = X( 2 ) ; Y( 23 ) = Y( 2 )
X( 3 ) = X( 2 ) - WTH4 / 2.0
Y( 3 ) = Y( 2 ) - WTH4 / 2.0
YZERO3 = Y( 3 )
DO 25 I = 3, 11
X( 1 ) = X( 3 )
X( 25-I ) = X( 3 ) + WTH4
Y( 1 ) = Y( 25-I ) = Y( 3 ) - AXLTH4 * FLOAT( I-3 ) / 8.0
25 CONTINUE
X( 12 ) = X( 11 ) - 1.2
X( 13 ) = X( 14 ) + 1.2
Y( 12 ) = Y( 13 ) = Y( 11 ) - 0.3
YZER11 = Y( 11 )
DO 100 I = 2, 23
X( 1 ) = GDTOCM( X( 1 ) ) ; Y( 1 ) = GDTOCM( Y( 1 ) )
100 CONTINUE
DO 150 I = 2, 23
LINK4( I ) = IPACK( X( 1 ), Y( 1 ), 1 )
150 CONTINUE
LINK4( 24 ) = 0
CALL GRAPHO( IDEV, LNK4( 2 ), 24, 7, IER )
IF( IER .NE. 0 ) OUTPUT( 101, IER, BAR4-1 )
DO 275 I = 1, 22
275 CONTINUE
IF( JA .EQ. 1 ) RETURN
500 DO 550 I = 2, 23
X( 1 ) = CMTOGD( X( 1 ) ) ; Y( 1 ) = CMTOGD( Y( 1 ) )
550 CONTINUE

```



```

000000470
000000480
000000490
000000500
000000510
000000520
000000530
000000540
000000550
000000560
000000570
000000580
000000590
000000600
000000610
000000620
000000630
000000640
000000650
000000660
000000670
000000680
000000690
000000700
000000710

Y(3) = Y(22) = YZERO3 + Q4(JA)*SF4
Y(23) = Y(2) + 0.2
Y(11) = Y(14) = YZER11 - Q7(JA)*SF4
Y(12) = Y(13) = Y(11) - 0.3
DIFF = ABS(Y(3) - Y(11)) - AXLTH4
DO 560 I = 4 10
Y(I) = Y(25-I) = Y(3) - (FLOAT(I-3) * (Y(3)-Y(11)) / 8.0)
CONTINUE
DO 575 I = 4 7
AMP(I) = DIFF*SIN(FLOAT(I-3) * PI/8;0) * SF4A
IF(AMP(I).GT.0.0) = AMP(I) = WTH4/2.0
X(I) = X(14-I) = X(3)+AMP(I)
X(25-I) = X(I+11) = X(I) + WTH4 - 2.0 * AMP(I)
CONTINUE
DO 600 I = 2,23
X(I) = GDTOCM(X(I)) ; Y(I) = GDTOCM(Y(I))
LINK4(I) = IPACK(X(I),Y(I),1)
CONTINUE
LINK4(2) = IPACK(X(2),Y(2),0)
LINK4(24) = 0
CALL GRAPHO(IDEV,LINK4,24,IER)
IF(IER.NE.0) OUTPUT(101) IER, BAR4-2
RETURN
END
560
575
600

```



```

SUBROUTINE CAM( NCAM, ICAM, RADIUS, AMP, JA, RADCOSS, RAD SIN)
DIMENSION ICAM( NCAM ), X( 97 ), Y( 97 ), RAD COS( 97 ), RAD SIN( 97 )
COMMON AXLTH1, AXLTH2, AXLTH3, AXLTH4
COMMON WTH1, WTH2, WTH3, WTH4
COMMON PIGDIR( 20 ), ITDIR( 40 )
COMMON XA, XB, YB, XC, XD, YE, XF, YF, XH, YH, XS, XP, YP
COMMON Q1( 180 ), Q2( 180 ), Q3( 180 ), Q4( 180 ), Q5( 180 ), Q6( 180 ), Q7( 180 )
COMMON Q8( 180 ), Q9( 180 ), QMEGA( 180 )
COMMON XCAM, YCAM
COMMON CMTOGD( A ) = A / 0.2000 * A - 1.0000
COMMON GDTOCN( A ) = 0.2000 * A - 1.0000
COMMON X( 1 ) = 0.0
COMMON X( 1 ) = NE * 1
COMMON ICAM( 1 ) = IHEAD( 0, 10 )
COMMON CROSS = 0.3
COMMON XCAM = XE - AMP
COMMON XC( 2 ) = XCAM + CROSS
COMMON XC( 3 ) = XCAM - CROSS
COMMON YCAM = YD - 0.6 - AXLTH1 - RADIUS + YS
COMMON Y( 2 ) = YCAM + CROSS ; Y( 3 ) = YCAM - CROSS
COMMON X( 4 ) = X( 5 ) = Y( 4 ) = Y( 5 ) = 0.0
DO 30 I = 1, 3
DO 30 X( I ) = GDTOCM( X( I ) ) ; Y( I ) = GDTOCM( Y( I ) )
CONTINUE
30 XCAM = GDTOCM( XCAM ); YCAM = GDTOCM( YCAM )
ICAM( 2 ) = IPACK( X( 2 ), YCAM, 0 )
ICAM( 3 ) = IPACK( X( 3 ), YCAM, 1 )
ICAM( 4 ) = IPACK( XCAM, Y( 2 ), 0 )
ICAM( 5 ) = IPACK( XCAM, Y( 3 ), 1 )
CONTINUE
55 IF( JA .EQ. 1 ) XCAM = CMTOGD( XCAM ) ; YCAM = CMTOGD( YCAM )
      XE = XCAM + COS( COMEGA( JA ) )
      YE = Q9( JA ) + YCAM
      X( 6 ) = RADIUS + XE ; Y( 6 ) = YE
      DC 200 I = 6, NCAM-1
      X( I+1 ) = RAD COS( I+1 ) + XE
      Y( I+1 ) = GDTOCM( X( I ) ) ; Y( I ) = GDTOCM( Y( I ) )
      IF( I .EQ. 6 ) ICAM( 6 ) = IPACK( X( 6 ), Y( 6 ), 0 ); GO TO 200
      ICAM( I ) = IPACK( X( I ), Y( I ), 1 )
CONTINUE
      ICAM( NCAM ) = 0
      CALL GRAPHO( IDEV, ICAM, NCAM, 1, IER )
      RETURN
END

```



```

SUBROUTINE DISPLAY(CONFIG, K1105, K1109(3), K1117(5), K1101(5); CONFIG(180)) 000000010
DIMENSION I101(7), I104(8), I105(3), I109(7), I117(5); BLNK 000000020
INTEGER ROTATE, CONFIG, YES, BLNK 000000030
COMMON AXLTH1, AXLTH2, AXLTH3, AXLTH4 000000040
COMMON WTH1, WTH2, WTH3, WTH4 000000050
COMMON PI, RAD, DEG, DIR(40) 000000060
COMMON IGDIR(20), IGDIR(40) 000000070
COMMON XA, YA, XB, YB, XC, YC, XD, YD, XE, YE, XF, YF, XH, YH, XS, YS, XP, YP 000000080
COMMON Q1(180), Q2(180), Q3(180), Q4(180), Q5(180), Q6(180), Q7(180) 000000090
COMMON Q8(180), Q9(180), QNEGA(180) 000000100
COMMON XCAM, YCAM 000000110
COMMON DATA NULL/777777B/, BLINK/4H /, YES/4HYES /, NO/4HNO /
NAMELIST IER, 1502, 1503, 1504, 1505, 1506, 1507, 1508) CONFIG(K)
CONTINUE
1501 CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 28, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 30, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 32, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
GO TO 1509
CONTINUE
1502 CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 28, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 30, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 32, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
GO TO 1509
CONTINUE
1503 CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 26, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 28, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER
CALL TEXTTO((IDEV, NULL, 1) 30, 30, 2, 3, IER)
IF(IER .NE. 0) OUTPUT(101) IER

```



```

IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,NULL,1,32,30,2,3,IER)
CONTINUE
1504 GO TO 1509
CALL TEXTO(IDEV,NULL,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1109,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1109,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1107,1,28,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,NULL,1,30,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1117,1,32,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
GO TO 1509
CONTINUE
1505 CALL TEXTO(IDEV,NULL,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1109,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,NULL,1,28,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1104,1,30,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,NULL,1,32,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
GO TO 1509
CONTINUE
1506 CALL TEXTO(IDEV,NULL,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1109,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,NULL,1,28,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1104,1,30,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1117,1,32,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
GO TO 1509
CONTINUE
1507 CALL TEXTO(IDEV,NULL,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1109,1,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1107,1,28,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXTO(IDEV,1104,1,30,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101) IER

```


00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030
00001040
00001050
00001060
00001070
00001080
00001090
00001100

CONTINUE
CALL TEXTO(IODEV,NULL,101,32,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101,32,30,2,3,IER)
GO TO 1509

1508 CONTINUE
CALL TEXTO(IODEV,NULL,101,26,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101,109,26,30,2,3,IER)
CALL TEXTO(IODEV,I09,I01,28,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101,107,28,30,2,3,IER)
CALL TEXTO(IODEV,I04,I08,30,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101,104,30,30,2,3,IER)
CALL TEXTO(IODEV,I17,I15,32,30,2,3,IER)
IF(IER.NE.0) OUTPUT(101,32,30,2,3,IER)

1509 CONTINUE
RETURN


```

00000010
00000020
00000030
00000040
00000050
00000060
00000070
00000080
00000090
00000100
00000110
00000120
00000130
00000140
00000150
00000160
00000170
00000180
00000190
00000200
00000210
00000220
00000230
00000240
00000250
00000260
00000270
00000280
00000290
00000300
00000310
00000320
00000330
00000340
00000350
00000360
00000370
00000380
00000390
00000400
00000410
00000420
00000430
00000440
00000450
00000460

SUBROUTINE INITIAL(CONFIG, IRRAY1, IRRAY2, IRRAY3, IRRAY4, IRRAY5, IRRAY6, IRRAY7, IRRAY9,
1   IRRAY10, ISPRNG)
2   INTEGER CONFIG, YES, BLNK
      DIMENSION IRRAY1(97), IRRAY2(18), IRRAY3(13), IRRAY4(32)
      DIMENSION IRRAY5(24), IRRAY6(36), IRRAY7(24), ISPRNG(9), IRRAY9(49)
      DIMENSION IRRAY10(49), IRRAY10(7)
      DIMENSION I106(12), I107(11), I108(7), I109(7), I110(2)
      DIMENSION I111(5), I112(9), I113(8), I114(18), I115(1), I116(10)
      COMMON CONFIG(180), I117(5)
      COMMON AXLTH1, AXLTH2, AXLTH3, AXLTH4
      COMMON WTH1, WTH2, WTH3, WTH4
      COMMON PI, RAD, DEG, IDEV
      COMMON IGDIR(20), ITDIR(40)
      COMMON XA, YA, XB, YB, XC, YC, XD, YD, XE, YE, XF, YF, XH, YH, XS, YS, YP
      COMMON Q1(180), Q2(180), Q3(180), Q4(180), Q5(180), Q6(180), Q7(180)
      COMMON Q8(180), Q9(180), Q10(180), QMEGA(180)
      COMMON XCAM, YCAM, ZCAM
      DATA NULL/7777777B/, BLNK/4H/, YES/4HYES/, NO/4HNNO/
      NAMELIST IER
      FORMAT(101, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120)
      FORMAT(101) CAM ACTUATED VALVE TRAIN
      FORMAT(106) THIS PROGRAM SIMULATES AN ELASTIC LINK MODEL
      FORMAT(107) OF AN AUTOMOBILE CAM ACTUATED VALVE TRAIN
      FORMAT(108) LT. CHARLES W. GNILKA USN
      FORMAT(109) VILLANOVA UNIVERSITY 1964
      FORMAT(110) ADVISOR
      FORMAT(111) DAVID SALINAS, PHD
      FORMAT(112) PROFESSOR OF MECHANICAL ENGINEERING
      FORMAT(113) U.S. NAVAL POSTGRADUATE SCHOOL
      FORMAT(114) FORMER ADVISOR: RICHARD C. WINFREY, PHD
      FORMAT(115) BY
      FORMAT(116) NAVAL CIVIL ENGINEERING LAB, PORT HUENEME, CALIF.
      FORMAT(117) WITH CLEARANCES
      IF(IDEV.EQ.1) OUTPUT(6) = AGT-1
      IF(IDEV.EQ.2) OUTPUT(6) = AGT-2
      OUTPUT(101) IDEV, INPUT CHANGES;
      DO 10 JA=1,20
      10 IGDIR(JA)=0
      CALL DGINIT(IDEV, IGDIR, 20, IER)
      IF(IER.NE.0) OUTPUT(101) IER
      CALL DTINIT(IDEV, ITDIR, 40, IER)
      IF(IER.NE.0) OUTPUT(101) IER
      INITIALIZE TEXTI BLOCK
      CALL TEXTO(IDEV, NULL, 1, 39, 46, 2, 3, IER)
      ENCODE(28, 101, 1101)

```



```

000000470
000000480
000000490
000000500
000000510
000000520
000000530
000000540
000000550
000000560
000000570
000000580
000000590
000000600
000000610
000000620
000000630
000000640
000000650
000000660
000000670
000000680
000000690
000000700
000000710
000000720
000000730
000000740
000000750
000000760
000000770
000000780
000000790
000000800
000000810
000000820
000000830
000000840
000000850
000000860
000000870
000000880
000000890
000000900
000000910
000000920
000000930
000000940

ENCODE(48, 106, 1106)
ENCODE(44, 107, 1107)
ENCODE(28, 108, 1108)
ENCODE(28, 109, 1109)
ENCODE(20, 110, 1110)
ENCODE(32, 111, 1111)
ENCODE(36, 112, 1112)
ENCODE(32, 113, 1113)
ENCODE(44, 114, 1114)
ENCODE(40, 115, 1115)
ENCODE(20, 116, 1116)
CALL(TEXTO(IDEV, PUT(110, 7, 1, 13, 3, 3, IER))
IF(IER.NE.0) OUTPUT(101, 5, 1, 2, 3, IER)
CALL(TEXTO(IDEV, PUT(101, 12, 5, 1, 2, 3, IER))
IF(IER.NE.0) OUTPUT(101, 11, 8, 6, 2, 3, IER)
CALL(TEXTO(IDEV, PUT(101, 15, 14, 45, 2, 3, IER))
IF(IER.NE.0) OUTPUT(101, 17, 25, 2, 3, IER)
CALL(TEXTO(IDEV, PUT(101, 7, 19, 37, 1, 3, IER))
IF(IER.NE.0) OUTPUT(101, 7, 19, 37, 1, 3, IER)
CALL(TEXTO(IDEV, PUT(101, 2, 24, 43, 2, 3, IER))
IF(IER.NE.0) OUTPUT(101, 5, 27, 29, 2, 3, IER)
IF(IER.NE.0) OUTPUT(101, 1, 29, 31, 1, 3, IER)
IF(IER.NE.0) OUTPUT(101, 9, 29, 31, 1, 3, IER)
IF(IER.NE.0) OUTPUT(101, 8, 30, 33, 1, 3, IER)
IF(IER.NE.0) OUTPUT(101, 14, 11, 33, 30, 1, 3, IER)
IF(IER.NE.0) OUTPUT(101, 16, 10, 34, 33, 1, 3, IER)
IF(IER.NE.0) OUTPUT(101, 17, 5, 11, 27, 2, 3, IER)
PAUSE TO READ TITLE PAGE
CALL(TEXTR(IDEVNUL, 139, 46, 2, 3, IER)
IF(IER.NE.0) OUTPUT(101, 97, 1, 0, 0)
IF(MOD(ITDIR(1), 8) .EQ. 0) GO TO 12
CALL(TEXTO(IDEVNUL, 139, 46, 2, 3, IER)
IF(IER.NE.0) OUTPUT(101, 1, 18, 0, 0)
DO 20 JA = 1, 97
20 IRRAY1(JA) = 1, 0
DO 21 JA = 1, 18
21 IRRAY2(JA) = 0, 0

```

12

C


```

000000950
000000960
000000980
000001000
000001010
000001020
000001030
000001040
000001050
000001060
000001070
000001080
000001090
000001100
000001110
000001120
000001130
000001140
000001150
000001160
000001170
000001180
000001190
000001200
000001210
000001220
000001230
000001240
000001250
000001260
000001270
000001280
000001290
000001300
000001310
000001320
000001330
000001340
000001350
000001360
000001370
000001380
000001390
000001400
000001410
000001420

DO 22 JA = 1,1
22 IRRAY3(JA) = 0,0
DO 23 JA = 1,32
23 IRRAY4(JA) = 0,0
DO 24 JA = 1,24
24 IRRAY5(JA) = 0,0
DO 25 JA = 1,36
25 IRRAY6(JA) = 0,0
DO 26 JA = 1,24
26 IRRAY7(JA) = 0,0
DO 27 JA = 1,9
27 ISPRNG(JA) = 0,0
DO 28 JA = 1,49
28 IRRAY9(JA) = 0,0
DO 29 JA = 1,49
29 IRRAY10(JA) = 0,0
      IHEAD(0,10)
IRRAY1(1) = IHEAD(0,10)
IRRAY2(1) = IHEAD(0,10)
IRRAY3(1) = IHEAD(0,10)
IRRAY4(1) = IHEAD(0,10)
IRRAY5(1) = IHEAD(0,10)
IRRAY6(1) = IHEAD(0,10)
IRRAY7(1) = IHEAD(0,10)
IRRAY8(1) = IHEAD(0,10)
IRRAY9(1) = IHEAD(0,10)
IRRAY10(1) = IHEAD(0,10)
      IDDEV,IRRAY1,97,1,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY2,18,2,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY3,13,3,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY4,32,4,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY5,24,5,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY6,36,6,IER)
CALL GRAPHO(IDEV,IRRAY7,24,7,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,ISPRNG,98,IER)
IF(IER.NE.0) CPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY9,49,9,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
CALL GRAPHO(IDEV,IRRAY10,49,10,IER)
IF(IER.NE.0) OUTPUT(101,IER,IERR)
RETURN
END

```



```

SUBROUTINE JOINT(NCIR, ICIR, RADIUS, ANGLE, COS( JA), RAD SIN( 75)
DIMENSION AXLTH1, AXLTH2, AXLTH3, AXLTH4
COMMON WTH1, WTH2, WTH3, WTH4
COMMON PI, RAD, DEG, IDEV
COMMON IDIR(20), IDIR(40)
COMMON XA, YA, XB, YB, XC, XD, YE, YF, YG, XH, YH, XS, XP, YP
COMMON Q1(180), Q2(180), Q3(180), Q4(180), Q5(180), Q6(180), Q7(180)
COMMON Q8(180), Q9(180), QMEGA(180)
COMMON XCAM, YCAM
GDTOCM(A) = 0.2000*A - 1.0000
NAMELIST IER
IF(JA .NE. 1) GO TO 35
ICIR(1) = IHAD(1,10)
X(1)=Y(1)=0.0
THETA = 0.0
DTHETA= ANGLE * RAD
DO 25 I=3, NCIR
THETA = THETA + DTHETA
RAD COS(I) = RADIUS*COS(THETA)
RAD SIN(I) = RADIUS*SIN(THETA)
CONTINUE
35 X(2) = RADIUS + XA ; Y(2) = YA
DO 100 I=3, NCIR
X(I) = RAD COS(I) + XA
Y(I) = RAD SIN(I) + YA
CONTINUE
100 OUTPUT(6) * JOINT CALLED*
      DO 200 I = 2, NCIR
      WRITE(6,99) I, X(I), Y(I)
      FORMAT(6,99)
      X(I) = GDTOCM(X(I), Y(I), 2E15*4)
      IF(I.EQ.2) ICIR(2)=IPACK(X(2), Y(2), 0); GO TO 200
      ICIR(I)=IPACK(X(I), Y(I), 1)
      CONTINUE
      CALL GRAPHO(IDEV, ICIR, NCIR, 7+K, IER)
      IF(IER.NE.0) OUTPUT(101) IER
      RETURN
END

```



```

SUBROUTINE MIDBAR(MIDCIR,IRRAY9)
DIMENSION AXLTH1,AXLTH2,AXLTH3,AXLTH4
COMMON WTH1,WTH2,WTH3,WTH4
COMMON PI,RAD,D2,E2,IDEV
COMMON IGDIR(20),ITDIR(40)
COMMON XA,YA,XB,YB,XC,YC,XD,YD,XE,YE,XF,YF,XH,YH,XG,YG,XS,YS,XP,YP
COMMON Q1(180),Q2(180),Q3(180),Q4(180),Q5(180),Q6(180),Q7(180)
COMMON Q8(180),Q9(180),OMEGA(180)
COMMON XCAM, YCAM
NAMELIST /IER/
GDTOCM(A) = 0.2000*A -.1.0000
MIDCIR(1) = IHEAD(0,10)
X(1)=Y(1)=0.0 ; K = 6 ; JA = 1
ANGLE = 9.0 ; RADIUS = 0.25
XA = XB
CALL JOINT(49,IRRAY9, RADIUS, ANGLE,K, JA)
DO 100 I = 1,49
IRRAY9(I) = 0
CONTINUE
100 X(2) = XB ; Y(2) = YB
X(3) = XB + .5 ; Y(3) = Y(.5)
X(4) = X(.5) - 1.0 ; Y(4) = Y(.5)
X(5) = X(.5) ; Y(5) = Y(.5)
X(6) = X(.5) ; Y(6) = Y(.5)
DO 150 I = 1,11
X(6+I) = X(5+I) -.1
CONTINUE
150 DO 160 I = 7,17,2
Y(I) = Y(.6) -.1
Y(I+1) = Y(.6)
CONTINUE
160 DO 200 I = 2,17
X(I) = GDTOCM( X(I) ) ; Y(I) = GDTOCM( Y(I) )
IF(I.EQ.2) MIDCIR(I) = IPACK(X(I),Y(I),1)
MIDCIR(I) = IPACK(X(I),Y(I),1)
CONTINUE
200 DO 250 I = 8,18,2
MIDCIR(I) = IPACK(X(I),Y(I),0)
CONTINUE
250 MIDCIR(18) = 0
CALL GRAPHO( IDEV,MIDCIR, 18,2,IER )
IF(IER.NE.0) OUTPUT(101) IER
RETURN
END

```

X


```

SUBROUTINE READ(NTIME, I104, XZERO, AMP, RPM, WN, ISEAT,
               PIN28, CONFIG, DIFF)
1  INTEGER CONFIG, YES, BLNK
DIMENSION I104(8), I105(3), CONFIG(180)
COMMON AXLTH1, AXLTH2, AXLTH3, AXLTH4
COMMON WTH1, WTH2, WTH3, WTH4
COMMON PI, RAD, DEG, IDEV
COMMON IGDIR(20), ITDIR(40)
COMMON XA, YA, XB, YB, XC, XD, YE, XF, YF, XH, YH, XS, YS, XP, YP
COMMON Q1(180), Q2(180), Q3(180), Q4(180), Q5(180), Q6(180), Q7(180)
COMMON Q8(180), Q9(180), UMEGA(180)
COMMON XCAM, YCAM
DATA NULL/7777777B/, BLNK/4H      /, YES/4HYES /, NO/4HNO /
NAMELIST IER
FORMAT(5E15.7)
31 FORMAT(5E15.7, 13)
40 FORMAT(4X, 13, 2X, 9(2X,1PE10:3), 6X, 12)
45 FORMAT(2X, *DEGREES, 6X, *Q1, 10X, *Q2,
         1          *Q4, 10X, *Q5, 10X, *Q6, 10X, *Q7, 10X, *Q8, 8X,
         1          *OMEGA, 7X, *CONFIG, //)
104 FORMAT(*LISTING OF COORDINATES DESIRED *)
105 FORMAT(*COMMAND: *)
5012 FORMAT(1*, 20X, 20A4)
5013 FORMAT(0*, 20X, 20A4)
5014 FORMAT(18A4)
5015 FORMAT(5E10.4, 2A4)
5016 FORMAT(0*, XZERO = E10.4, AMP = E10.4, RPM = E10.4,
         1          NAT, FREQ = 3PE10.1, VALVE SEAT, 1A4,
         2          COORD. 2 AND 8 PINNED)
5017 FORMAT(0*, 10X, AT ZERO DEGREES -, CLEARANCE BETWEEN VALVE,
         1          AND VALVE SEAT IS , F7.3)
4  REWIND 4
      READ(4, 5014) (Q1(JA), JA=1, 18)
      READ(4, 5014) (Q2(JA), JA=1, 18)
      READ(4, 5015) XZERO, AMP, RPM, WN, CLEAR, ISEAT, PIN28
      WRITE(6, 5012) (Q1(JA), JA=1, 18)
      WRITE(6, 5013) (Q2(JA), JA=1, 18)
      WRITE(6, 5016) XZERO, AMP, RPM, WN, ISEAT, PIN28
      DESIRE(COORDINATES PRINTED
      ENCODE(32, 104, 1104)
      ENCODE(12, 105, 1105)
      CALL TETO(IDEV, 1104, 8, 37, 17, 2, 3, IER)
      IF(IER.EQ.0) CUTPUT(101) IER
      CALL TETO(IDEV, 1105, 39, 23, 2, 3, IER)
      IF(IER.EQ.0) CUTPUT(101) IER
      CALL TETR(IDEV, NULL, 1, 39, 46, 2, 3, IER)

```

C


```

      IF( IER .NE. 0 ) OUTPUT(101) IER
      IF( MOD( ITDIR(1),8) .EQ. 0 ) GO TO 15
      CALL TEXTI( IDIR(1),3946,IER)
      IF( IER .NE. 0 ) OUTPUT(101) IER
      IPRINT = NO
      IF( IMODA .EQ. YES ) IPRINT = YES
      ERASE QUESTI( IDEV, NULL(1),3946,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),3946,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),3946,2,3,IER)
      IF( IER .NE. 0 ) OUTPUT(101) IER
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),3717,2,3,IER)
      IF( IER .NE. 0 ) OUTPUT(101) IER
      READ(4,30) DIFF
      READ(4,50) DIFF
      DO 1000 I = 1,NTIME
      READ(4,30) Q1(I),Q2(I),Q3(I),Q4(I),Q5(I),OMEGA(I),CONFIG(I)
      CONTINUE
      IF( IPRINT .EQ. NO ) GO TO 1250
      WRITE(6,45)
      DO 1200 I = 1,NTIME
      NDEG = IFIX( OMEGA(I)*DEG )
      WRITE(6,40) NDEG,Q1(I),Q2(I),Q3(I),Q4(I),Q5(I),Q6(I),Q7(I),
      1200 CONTINUE
      NULL PAGE
      CALL TEXTO( IDEV, OUTPUT(101),1,5,1,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),1,8,6,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),1445,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),1725,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),1937,1,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),2443,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),2729,2,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),2931,1,3,IER)
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),3033,1,3,IER)
      IF( IER .NE. 0 ) OUTPUT(101) IER
      CALL TEXTO( IDEV, OUTPUT(101),NULL(1),3330,1,3,IER)
      IF( IER .NE. 0 ) OUTPUT(101) IER
      15
      C

```



```
000000950
000000960
000000970
000000980
000000990
000001000

CALL TEXT0(IDEV, NULL, 1, 34, 33, 1, 3, IER)
IF(IER.NE.0) OUTPUT(101) IER
CALL TEXT0(IDEV, NULL, 1, 11, 27, 2, 3, IER)
IF(IER.NE.0) OUTPUT(101) IER
RETURN
END
```



```

SUBROUTINE SPRCTR(IISPRNG,ISEAT,DIFF)
DIMENSION X(13),Y(13),IISPRNG(I3),ISEAT(I3),DIFF
INTEGER YES,NO
COMMON AXLTH1,AXLTH2,AXLTH3,AXLTH4
COMMON WTH1,WTH2,WTH3,WTH4
COMMON PI,RAD,IDEV
COMMON IGDIR(20),ITDIR(40)
COMMON XA,YA,XB,YB,XC,YC,XD,YD,XE,YE,XF,YF,XH,YH,XS,YS,XP,YP
COMMON Q1(180),Q2(180),Q3(180),Q4(180),Q5(180),Q6(180),Q7(180)
COMMON Q8(180),Q9(180),OMEGA(180)
COMMON XCAM,YCAM
COMMON XES/4HYES/, NO/4HN0 /
NAMELIST IER
DATA YES/4HYES/, NO/4HN0 /
GDTOCM(A) = 0.2000*A - 1.0000
IISPRNG(1) = IDEV
X(1)=Y(1)=0.0
X(2)=XC;Y(2)=YC-.3
X(3)=XC;Y(3)=YC+.3
DO 100 I=1,100
Y(3+I)=Y(2+I)-:1*(1.0)**(I+2)
X(I+3)=X(I+2)-:1*(-1.0)**(I+2)
100 CONTINUE
DO 175 I=GDTOCM(X(1)) ; Y(I) = GDTOCM(Y(I))
175 CONTINUE
DO 200 I=2,10
IF(I.EQ.2) IISPRNG(I)=IPACK(X(I),Y(I),1)
200 CONTINUE
DO 250 I=5,9,2
IISPRNG(I)=IPACK(X(I),Y(I),0)
250 CONTINUE
IISPRNG(11)=0
CALL GRAPHO(IDEV,IISPRNG,11,IER)
IF(IER.NE.0) OUTPUT(101) IER,IISPRCTR,
IF(ISEAT.EQ.NO) DIFF=1.2
K=0
SIGN=+1.0
X(12)=XS
Y(2)=YH-WTH4/2.0-AXLTH4+DIFF+YP
Y(3)=Y(2)-0.4
Y(4)=Y(2)
DO 275 I=4,10,2
Y(I+1)=Y(I+2)=Y(I)-.1
275 CONTINUE
300 K=K+1

```



```

000000470
000000480
000000490
000000500
000000510
000000520
000000530
000000540
000000550
000000560
000000570
000000580
000000590
000000600
000000610
000000620
000000630
000000640
000000650
000000660
000000670
000000680
000000690
000000700
000000710
000000720
000000730
000000740
000000750
000000760
000000770
000000780
000000790
000000800
000000810

DO 350 I = 1,7
X(12-1) = X(13-1)*2 + SIGN*0.3
X(11-1) = X(12-1) + SIGN*0.1
CONTINUE
X(3) = X(11)
X(2) = X(4) + 0.3*SIGN
X(13) = Y(13) = 0.5
DO 375 I = 1,13
X(1) = GDTOCM(X(1))
IF(K.EQ.1) Y(I) = GDTOCM(Y(I))
CONTINUE
ISPRNG(2) = IPACK(X(2),Y(2),0)
ISPRNG(3) = IPACK(X(3),Y(3),1)
ISPRNG(4) = IPACK(X(4),Y(4),0)
ISPRNG(5) = IPACK(X(2),Y(2),1)
DO 400 I = 6,12
ISPRNG(1) = IPACK(X(I-1),Y(I-1),0)
ISPRNG(I+1) = IPACK(X(I),Y(I),1)
CONTINUE
ISPRNG(13) = 0
CALL GRAPHO(IDEV,ISPRNG,13,10+K,IER)
IF(IER.NE.0) OUTPUT(101)
X(12) = XP
SIGN = -1.0
IF(K.EQ.1) GO TO 300
RETURN
DO 500 I = 2, 13
ISPRNG(I) = 0
CONTINUE
CALL GRAPHO(IDEV,ISPRNG,13,11,IER)
IF(IER.NE.0) OUTPUT(101)
CALL GRAPHO(IDEV,ISPRNG,13,12,IER)
IF(IER.NE.0) OUTPUT(101)
RETURN
END

```


LIST OF REFERENCES

1. Winfrey, R. C., Dynamics of Mechanism With Elastic Links, Research and Development Division, Aerospace Group, Hughes Aircraft Co., Report No. JT 5180, March 1969.
2. Anderson, R. V., Dynamic Analysis of Cams, M. S., ME Thesis U.S. Naval Postgraduate School, 1971.
3. Thomson, W. T., Vibration Theory and Applications, 1st ed., Prentice-Hall, Inc., New Jersey 1965.
4. Newmark, N. M., "A Method of Computation for Structural Dynamics," ASCE, Journal of Engineering Mechanics Division, v. 85, No. EM3, p. 67-94, July 1959.
5. Scientific Data Systems, SDS- FORTRAN IV REFERENCE MANUAL, SDS, Santa Monica, California, 1966.
6. Scientific Data Systems, SDS-9300 Computer Reference Manual, SDS, Santa Monica, California, 1966.
7. Scientific Data Systems, SDS Real Time Monitor Reference Manual, SDS, Santa Monica, California, 1966.
8. Rubinstein, M. F., Matrix Computer Analysis of Structures, 1st ed., Prentice-Hall, Inc., 1965.
9. Hurty, W. C., and Rubinstein, M. F., Dynamics of Structures, 1st ed., Prentice-Hall, Inc., 1966.
10. International Business Machines Corporation Form C28-6515-7, IBM System/360 FORTRAN IV Language, 1968.
11. International Business Machines Corporation Form H20, 0205-3, IBM System/360 Scientific Subroutine Package (360-CM-03X) Version III Programmers Manual, 1968.
12. IBM System/360 Operating System, FORTRAN IV (H) Programmer's Guide, Form C-28-6602-1, IBM Corp., Poughkeepsie, N.Y.
13. W. R. Church Computer Center, Users Manual, 1st ed., Rev. 6, March 1970.
14. Mitchell, D. B., "Cam Follower Dynamics", Machine Design, v. 22, p. 151-154, June 1950.
15. Rothbart, H. A., Cams, 1st ed., John Wiley & Sons, Inc., 1956.

16. Stoddart, D. A., "Polydyne Cam Design", Machine Design, v. 25, p. 121-135, January 1953.
17. Molian, S., "Use of Algebraic Polynomials in Design of Cams", The Engineer, v. 215, p. 352-354, 22 February, 1963.
18. Barkan, P., and McGarrity, R. V., "A Spring-Actuated, Cam-Follower System: Design Theory and Experimental Results", Journal of Engineering for Industry, v. 87, p. 279-286, August 1965.
19. Gnilka, C. W., and Bowden, P. K., Dynamic Analysis of a Four Bar Mechanism Utilizing Interactive Computer Graphics, paper presented for M.E. Laboratory Project (ME-3400), Dept. of Mech. Eng., U.S. Naval Postgraduate School, June 1971.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3. Director, Computer Facility, Code 0211 Naval Postgraduate School Monterey, California 93940	1
4. Chairman, Department of Mechanical Engineering Code 59 Naval Postgraduate School Monterey, California 93940	1
5. Professor David Salinas, Code 59Zc Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	3
6. Dr. Richard C. Winfrey Naval Civil Engineering Laboratory Port Hueneme, California	3
7. Supervisor, Computer Laboratory Code Ec Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
8. Charles W. Gnilka, LT, USN Philadelphia Naval Shipyard Philadelphia, Pennsylvania 19112	5

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)	
REPORTING ACTIVITY (Corporate Author) Naval Postgraduate School Monterey, California	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

REPORT TITLE
Modeling and Analysis of an Elastic Mechanism With Clearances

SUPPLEMENTARY NOTES (Type of report and inclusive dates)

Master's Thesis; December 1971

(ORIGINATOR'S NAME) (First name, middle initial, last name)

Charles W. Gnilka

REPORT DATE December 1971	7a. TOTAL NO. OF PAGES 342	7b. NO. OF REFS 19
CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)	
OBJECT NO.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	

DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

APPENDIXES AND RELATED MATERIALS SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940
---	---

ABSTRACT

The classical approach for the analysis of linkages has been to assume rigid elements during operation. As the operational speeds of mechanisms increase, this assumption creates larger errors in the analysis. Thus a better model of a machine that considers elastic deformations will aid in more efficient design as well as provide improved accuracy and performance.

This thesis presents a simulation for the dynamic response of an elastic link model with three clearances. The simulation has been developed for an automobile, cam actuated valve train. The computer program is coded in FORTRAN IV, for an IBM-360 computer and is included as a portion of this work. The capability to visually observe the dynamic action of the model is included by a graphic display routine. This routine is implemented in SDS FORTRAN IV for a SDS-9300 computer interfaced with an ADAGE Graphics Display Terminal, Model 10.

Sample problems for various cam speeds of interest are included utilizing graphs and photos from the graphic display.

UNCLASSIFIED

Security Classification

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Lastic Mechanism am Actuated Valve Train omputer Graphics learance ynamic Modeling mputer Simulation						

BINDERY

Thesis

133468

G522 Gnilka

c.1 The modeling and analysis of an elastic mechanism with clearances.

BINDERY

Thesis

133468

G522 Gnilka

c.1 The modeling and analysis of an elastic mechanism with clearances.

thesG522
The modeling and analysis of an elastic



3 2768 002 02960 5
DUDLEY KNOX LIBRARY